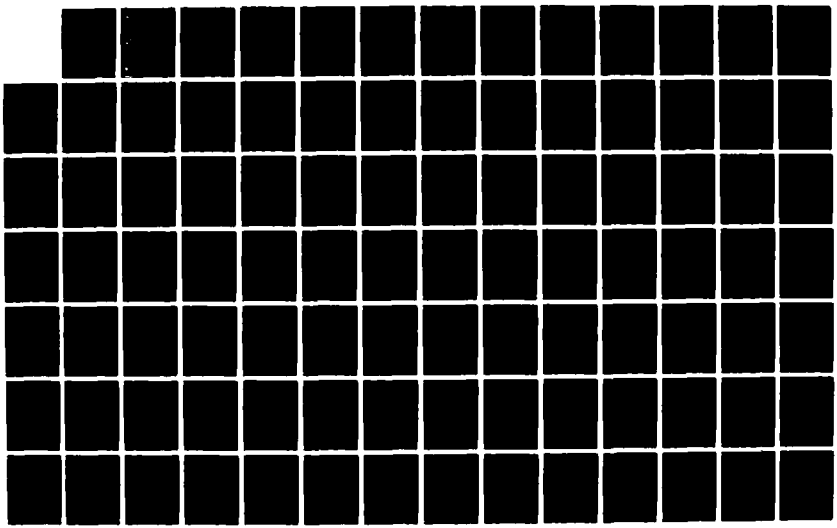
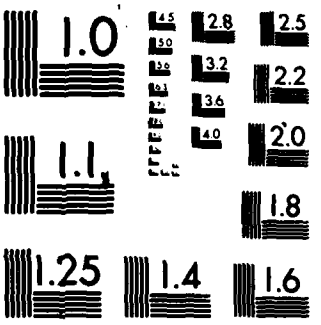
AD-A138 311 POINTING AND TRACKING OF PARTICLE BEAMS(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING W L ZICKER DEC 83 AFIT/GE/EE/83-73 1/2

UNCLASSIFIED NL





POINTING AND TRACKING OF PARTICLE BEAMS

THESIS

AFIT/GE/EE/83-73 William L. Zicker
1st Lt USAF

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/EE/83-73

POINTING AND TRACKING OF PARTICLE BEAMS

THESIS

AFIT/GE/EE/83-73 William L. Zicker
lst Lt USAF

Approved for public release; distribution unlimited

AFIT/GE/EE/83-73

POINTING AND TRACKING OF PARTICLE BEAMS

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air Training Command
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

William L. Zicker
1st Lt USAF
Graduate Electrical Engineering

December 1983

A-1

Preface

This thesis was under the sponsorship of the Air Force Weapons Laboratory at Kirtland Air Force Base in New Mexico. It is a theoretical study to determine the feasibility of using a Multiple Model Adaptive Estimator based on space-time point process observations, as developed by Capt. David E. Meer, in a closed loop control system to direct a particle beam. This study involves a knowledge of optimal filter theory, optimal control theory, and software simulation techniques that made this study fascinating.

I would like to thank my thesis advisor, Dr. Peter S. Maybeck, for his guidance and expertise, and especially his patience with my efforts at this, my first major piece of technical writing.

William L. Zicker

Contents

Figures

Abstract

A problem is considered to determine the feasibility of using a Multiple Model Adaptive Estimator (MMAE) based on space-time point process observations, as part of a control loop. The estimator tracks the centroid of a one-dimensional Gaussian-shaped source of individual photo-electron events. The centroid is assumed to move dynamically as a first order Gauss-Markov process. Estimator performance for two implementations (using "best half" versus "merge" rationale for limiting the number of individual filters in the MMAE structure) is described in terms of steady-state root mean square (RMS) error evaluated as a function of six important system parameters. Estimates of the beam centroid are used by a regulator based on a Linear system and Quadratic cost criteria (LQ) to determine the effectiveness of the method. For tracking purposes, a real-world target and a Kalman filter are provided based on a first order Gauss-Markov process model, and the controller is tested in this environment. Results indicate that the estimator provides a viable means of controlling the position of the centroid.

POINTING AND TRACKING OF PARTICLE BEAMS

I Introduction

Problem

The future use of particle beams as weapons depends upon adequate methods existing for directing the beam at its target. Therefore, it is necessary to look at the problem of directing particle beams to a specific point in space.

Motivation. The propagation of a particle beam is governed by, in addition to the generating and directing fields, internally generated fields. These interactions make some sort of feedback controller necessary in attempts to direct a particle beam.

However, in order to use feedback control, it is necessary to determine the location of the beam relatively accurately in real time. One method which has been proposed is to direct a laser beam through the beam path and observe the photons resulting from interactions between the particles and the laser, to determine the beam location.

Until recently there has not been a method for estimating the beam position in the presence of the point process noises arising from spontaneous photon emission in the beam and the spontaneous generation of electrons within the photodetector. However, Capt. David Meer has just completed a PhD dissertation in which he developed an adaptive filter that can specifically

address this problem (Ref 22).

Statement. It is proposed to use the estimates of beam position derived from the estimator developed by Capt. Meer in conjunction with a controller derived by invoking assumed certainty equivalence (Ref 17), to be described more fully in the next section on the problem scope. One of the primary questions is to determine if the filter developed by Capt. Meer can be used in this manner or if it is so sensitive to variations from its assumed model that it may not be usable in a realistic engineering situation. Also, information about the target will be coming in through some estimation process and it is important to determine if the two filtering methods can work together, or if there is some destructive interaction arising from the difference in their basic assumptions.

Scope. The scope of this effort will be to look at the performance of Capt. Meer's filter as part of a feedback controller. The effort will encompass both the regulator and tracker problems. The model for the actual particle beam is kept simple and essentially the same as that assumed by filter.

For this effort the centroid of the particle beam will be considered to be adequately described by a first order Gauss-Markov process in one dimension. Only one dimension will be used to describe the beam location, an x coordinate on the photodetector focal plane, in order to keep the filter and controller simple and low-dimensional for this feasibility demonstration.

The beam is assumed to be operating close to an equilibrium, with only small perturbations so that linear models and techniques may be considered adequate. Also, the measurements of the beam are assumed to be taken near the device so that there are no delays in the system response due to beam propagation time.

The feedback controller will be synthesized using the assumption of certainty equivalence, in which an optimal deterministic controller is synthesized on the basis that there is access to the actual states rather than the best estimates of them. Then this controller is in fact provided the output of the state estimator. This concatenation has proven to be the optimal solution under LQG (linear system, Quadratic cost for optimality criterion specification, and Gaussian noise models) and a good controller synthesis assumption for extended Kalman filter applications and other stochastic processes (Refs 7, 9, 17, Vol III, 35), so, unless evidence to the contrary arises from our study, it seems to be a good assumption to make.

Finally, in working the tracker problem, the target model proposed is that the position (plus unmodeled effects in the truth model) of the target is a first order Gauss-Markov process (Refs 17, 21). This model is chosen rather than a more realistic higher order process (Ref 21), because we are only interested in the feasibility of the method and do not wish to become involved in more complex implementation issues.

Background Literature

Poisson Processes. Because the photons which provide the measurements are generated by spontaneous emission, a quantum mechanical process (Refs 8, 27), the measurement update times have a random rather than systematic distribution. The intervals between events in this case are adequately described by a Poisson distribution. Snyder and Fishman have developed an estimation scheme which yields good results in the absence of noise events (Refs 10, 33, 34). However, Santiago has shown that the performance is seriously degraded in the presence of events which are not originated in the source modeled by the estimation scheme (Ref 29).

Multiple Model Adaptive Estimation. Capt. Meer has demonstrated that good performance under these conditions may still be retained (Ref 22) through a variation of Multiple Model Adaptive Estimation (MMAE). There is extensive literature available on MMAE as it has been previously applied to Gaussian noise processes in references 1, 2, 4, 6, 11, 16, 17, Vols II and III, 23, 31, 37, and 38. There are two basic conditions which can drive the use of MMAE. The first is that the problem in question cannot be well described for all time by a single model. The second is that at least one parameter varies over such a wide range that adequate filter performance using a single value for the parameter over the range of operating conditions is precluded.

For such cases a set of estimators ("bank of filters"

which are Kalman filters in the linear Gaussian model case) is designed; one filter matched to each distinct model, or to a discretized set of parameters extending over a continuous range of values. The estimates from each filter are then probabilistically weighted and summed to obtain an overall estimate. There is also a body of literature on the decision theory underlying the weighting factors available in references 3, 12, 13, 14, 30, and 32. For the Gaussian case, weighting factors are calculated using any a priori statistical knowledge of the models that is available, and the residual time history.

What Capt. Meer did was to substitute Snyder and Fishman estimators into the bank of filters, each operating on different assumptions concerning the source of measurement events. He also derived a method of calculating the weighting factors necessary to summing the estimates. The performance with the presence of noise events was significantly better than that obtained by the unaided Snyder and Fishman estimator, though due to the use of the weighting factors, there was a loss in the ability to recover from bad initial estimates. This arises from the fact the filter places a very low weighting factor on measurements which are distant from the current estimate.

LQG Controller Design. Also important to this effort are the well developed techniques for designing optimal controllers for linear systems and for performing sensitivity

analyses (Refs 5, 28). In particular, the LQG regulator problem has been extensively covered in references 7, 9, 15, 17, Vol. III, and 35. The simplest method involving assumed certainty equivalence and the solution of the backward Riccati equation results in a requirement of full state feedback, which is not a computational difficulty for this problem as the number of states used is small and the system is completly observable.

As developed in Maybeck (Ref 17) Vol III, the tracker problem is seen to be merely an extension of the regulator. References 18, 19, 20, and 21 also yield some insight into issues concerning the implementation of tracker simulations.

Approach

The feasibility of this control scheme will be judged by computer simulation. As we are interested in a stochastic process, we desire to evaluate the statistical behavior of the process. To get this statistical information, we will use the Monte Carlo method to generate a close approximation to the process statistics. Because we will be dealing with point process statistics, the number of Monte Carlo simulations for the system will probably have to be on the order of a hundred rather than the fifteen or twenty considered adequate for more standard Gaussian stochastic processes (Refs 17, 22). The adequacy of the number of Monte Carlo runs will be ensured in Chapter V before the performance analyses are actually conducted.

This effort is in the nature of a first look at the feasibility of this scheme and not an exhaustive wringing out of a design that is actually to be implemented. Therefore, the software developed for the design will be made as flexible as possible so that future efforts to develop an actual implementation will only involve easily made changes to the models the simulation package uses.

Using first order linear models, a controller will be synthesized for the simple state regulator problem. Linear system/quadratic cost techniques will be used to design this controller, assuming access to the perfectly known states of the model.

The performance of the closed loop system incorporating filter and regulator will be analyzed to assess the root mean square (rms) error in pointing the beam. These results will be compared to the performance of a closed loop system using the reference model states (used for simulation of the real world environment) themselves, in order to assess the impact of the Meer filter on closed loop control system performance. The sensitivity of the system to changes in various parameters with the filter given perfect knowledge of the parameter values will be considered.

A similar approach will be taken in extending the system to cover the tracker problem with the addition of a truth model to represent target dynamics. This extension then requires an additional Kalman filter to estimate these additional var-

iables for the controller. In this phase, we will extend the comparisons to include the Meer filter without the Kalman filter and vice versa, as well as with both filters and with no filtering, in an effort to determine the relative contributions of the filters to closed loop system performance. Sensitivity analyses will also be run for this phase of the problem.

Summary of Remaining Chapters

In Chapter II, the signal-in-noise model is defined and the equations of the Snyder and Fishman estimator are shown. The structure of the hypothesis tree underflying the MMAE used in this problem is explained, including the calculation of the weighting factors. Finally, two methods of limiting the estimator's size are discussed.

In Chapter III, the controller design procedure used for the regulator and the tracker problems are explained. Also in this chapter the model for the target used in the tracker problem is presented as well as the tracking filter used.

In Chapter IV, the simulation design is laid out, including the simulation of the Poisson processes. Finally, the performance parameters used to evaluate the estimator and controller performances are explained.

In Chapter V, the results of the simulation are shown. There will be comparisons between the two methods of limiting the filter size shown in Chapter II. The performance of the regulator under various conditions will be examined. And fi-

nally the tracker performance results will be presented.

Chapter VI is devoted to conclusions derived from this work and recommendations concerning applications and further investigations of the estimator.

II Multiple Model Adaptive Estimator

The key to directing a neutral particle beam is to know where the beam is currently pointing. In this chapter, the full structure of the multiple model adaptive estimator developed by Meer is explained. The first section deals with the form of the estimator required by the Poisson nature of the measurement events. The next section deals with the measurement history "tree" which determines the number of estimators needed in the "bank of filters". The following section deals with how the estimates are melded by a probabilistically weighted sum into a single estimate. Also shown is how the a posteriori filter weights are used to limit the number of estimators from growing geometrically with each measurement event.

Poisson Nature of Discrete Events

The signal source consists of the set of particles in a volume of the neutral particle beam excited by a laser directed through it. The spontaneous decay of these particles from their excited states results in the emission of photons which are observed by a photodetector array. The time distribution of these photo-electron events may readily be modeled as Poisson (Refs 27, 33).

In developing their dynamic system state estimator, Snyder and Fishman modeled the signal events as a space-time

point process (Ref 33). Each photon detection event thus has associated with it a time of occurrence t and a spatial location $\bar{r}$ on the detector array. It is assumed that the excited particle density is such that the signal rate parameter λ_s describing the rate of occurrence of observation events at time t and location $\bar{r}$ can be expressed by:

$$\lambda_s(t,\bar{r},\underset{\sim}{\bar{x}}(t)) = \Lambda(t)\exp\{-[\bar{r}-\underline{H}(t)\underset{\sim}{\bar{x}}(t)]^T\underline{R}^{-1}(t) [\bar{r}-\underline{H}(t)\underset{\sim}{\bar{x}}(t)]/2\} \quad (1)$$

where: $\Lambda(t)$ is the amplitude of the rate density function; $\underline{H}(t)$ is an m x n projection matrix describing the relationship between the location of the beam centroid on the sensor array and the states $\underset{\sim}{\bar{x}}(t)$ of a shaping filter used to model the dynamic behavior of the centroid; $\underline{R}(t)$ is a symmetric positive definite matrix describing the beam width as a dispersion matrix, analogous to a covariance, about the beam centroid; $\underset{\sim}{\bar{x}}(t)$ is an n-dimensional Gaussian output of a linear stochastic differential equation describing the behavior of the beam centroid

$$\dot{\underset{\sim}{\bar{x}}}(t) = \underline{F}(t)\underset{\sim}{\bar{x}}(t) + \underline{G}\underset{\sim}{\bar{w}}(t) \quad (2)$$

where $\underset{\sim}{\bar{w}}(t)$ is a white Gaussian noise of unit strength which, in conjunction with the linearity of the equation, yields a Gauss-Markox state process $\underset{\sim}{\bar{x}}(t)$. This form of $\underset{\sim}{\bar{x}}(t)$ is used because it describes a large number of useful estimation

problems, and results in an estimator which is similar to a Kalman filter.

The noise induced photo-electron events are modeled as a space-time Poisson process with rate parameter $\lambda_n(t,\bar{r})$. The rate parameter is allowed to be dependent on a random process $\underset{\sim}{\theta}$, thus $\lambda_n(t,\bar{r})$ can also be a random process; however, in our application, it will not be.

The noise process is assumed to be statistically independent of the signal process and additive. The observed point process is thus composed of the sum of the two point processes. Since they are independent processes, the probability density function of their sum is the convolution of the individual density functions, and the characteristic function of the sum is the product of both processes' characteristic functions.

The characteristic function for the signal process is

$$\phi_s(\omega) = \exp[\lambda_s(t,\bar{r},\bar{x}(t))(e^{j\omega}-1)] \qquad (3)$$

and the characteristic function of the noise process is

$$\phi_n(\omega) = \exp[\lambda_n(t,\bar{r})(e^{j\omega}-1)] \qquad (4)$$

where $j = \sqrt{-1}$ only for these two expressions and where ω is the frequency domain variable (Ref 26).

The product of these two conditional characteristic functions is the characteristic function of the sum of the processes. The form of the product is that of a conditional

Poisson point process with rate parameter

$$\underset{\sim}{\lambda}(t,\bar{r},\bar{x}(t)) = \lambda_s(t,\bar{r},\underset{\sim}{\bar{x}}(t)) + \underset{\sim}{\lambda}_n(t,\bar{r}) \tag{5}$$

In the absence of noise events, the signal model results in the following estimator equations expressed in differential form

$$d\hat{\bar{x}}(t) = \underline{F}(t)\hat{\bar{x}}(t)dt + \int_{R^m} \underline{K}(t)[\bar{r}-\underline{H}(t)\hat{\bar{x}}(t)]\, N(dt \times d\bar{r}) \tag{6}$$

$$\begin{aligned} d\underline{\Sigma}(t) = \underline{F}(t)\underline{\Sigma}(t)dt + \underline{\Sigma}(t)\underline{F}^T(t)dt + \underline{G}(t)\underline{Q}(t)\underline{G}^T(t)dt \\ - \int_{R^m} \underline{K}(t)\underline{H}(t)\underline{\Sigma}(t)N(dt \times d\bar{r}) \end{aligned} \tag{7}$$

$$\underline{K}(t) = \underline{\Sigma}(t)\underline{H}^T(t)[\underline{H}(t)\underline{\Sigma}(t)\underline{H}^T(t) + \underline{R}(t)]^{-1} \tag{8}$$

where $\int_{R^m} N(dt \times d\bar{r})$ is a counting integral (Refs 10, 33) evaluated as

$$\int_{R^m} f(t,\bar{r})N(dt \times d\bar{r}) = \begin{cases} 0, & N_t = 0 \\ \sum_{i=1}^{N_t} f(t_i,\bar{r}_i) & N_t \geq 1 \end{cases} \tag{9}$$

this form has been shown by Santiago (Ref 29) to allow the separation of equations (6) and (7) into propagation and update equations as follows

$$d\hat{\bar{x}} = \underline{F}(t)\hat{\bar{x}}(t)dt \tag{10}$$

$$d\underline{\Sigma}(t) = \underline{F}(t)\underline{\Sigma}(t)dt + \underline{\Sigma}(t)\underline{F}^T(t)dt + \underline{G}(t)\underline{Q}(t)\underline{G}^T(t)dt \tag{11}$$

and the update equations

$$\hat{\bar{x}}(t_i^+) = \hat{\bar{x}}(t_i^-) + \underline{K}(t_i^-)(\bar{r}-\underline{H}(t_i^-)\hat{\bar{x}}(t_i^-)) \tag{12}$$

$$\underline{\Sigma}(t_i^+) = \underline{\Sigma}(t_i^-) - \underline{K}(t_i^-)\underline{H}(t_i^-)\underline{\Sigma}(t_i^-) \tag{13}$$

where t_i^- is a time immediately prior to an observed event, and t_i^+ is a time just after an observed event and its incorporation into the estimate, and $\underline{K}(t_i)$ is defined as in equation (8).

This estimator performs quite well in simulations that do not incorporate any noise events. However, real detectors both receive noise from the background and generate some spurious electrons even in the absence of any photons. These noise events have been shown by Santiago to degrade the accuracy of the estimator seriously (Ref 29), unless some means of identifying and rejecting these noise events is used.

Tree Development

Each observed point event results from either the underlying signal generation process or the noise generation process. The two possibilities for each observed point event lend themselves to the idea constructing all of the possible sequences of noise/signal events which could have produced the

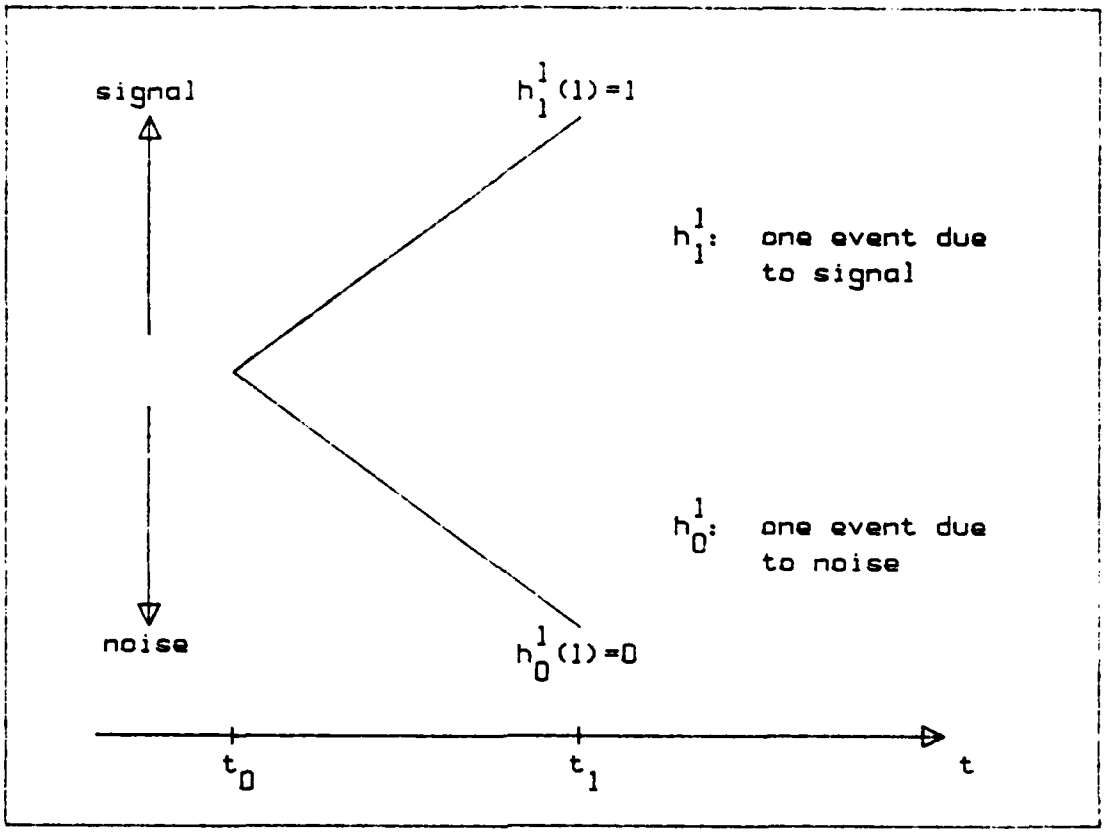


Figure 1. Hypothesis Sequences for One Measurement.

observed sequence. The following development of the hyothesis "tree" and the corresponding notation closely follows that presented by Meer (Ref 22:34-38) in his dissertation.

In this example we begin at t_0 when no events have yet been observed. Then the first point event is observed at time t_1 and location $\bar{r}_1$ and we have a measurement sequence of one data point for the observation interval. As this event could have been generated either by the signal process or by the noise process, two possible hypotheses for the measurement sequence naturally arise. These hypotheses are represented by the tree diagram in Figure 1.

A hypothesis sequence will be represented by $h_j^{N_t}$ where the subscript $j\varepsilon(0,1,\ldots,(2^{N_t})-1)$ denotes which sequence is identified and the superscript N_t is the total number of events observed. When an argument is present as in $h_j^{N_t}(i)$, $i=1,2,3,\ldots N_t$, the value of the sequence at time t_i is being referenced. Thus, a hypothesis sequence is written in the following form

$$h_j^{N_t} = \{h_j^{N_t}(1), h_j^{N_t}(2), \ldots h_j^{N_t}(N_t)\} \tag{14}$$

In Figure 1, the hypothesis that the event observed at t_1 was due to signal is denoted by the hypothesis sequence h_1^1, while the hypothesis that this event was due to noise is denoted by the sequence h_0^1. The sequence h_1^1 consists of the single entry

$$h_1^1 = \{h_1^1(1)\} = \{1\} \tag{15}$$

where $h_1^1(1)$ is the value of the sequence h_1^1 at time t_1. Values are assigned as

$$h_j^{N_t}(i) = \begin{cases} 1 : \text{event due to signal} \\ 0 : \text{event due to noise} \end{cases} \tag{16}$$

By the same method, the sequence h_0^1 is defined as

$$h_0^1 = \{h_0^1(1)\} = \{0\} \tag{17}$$

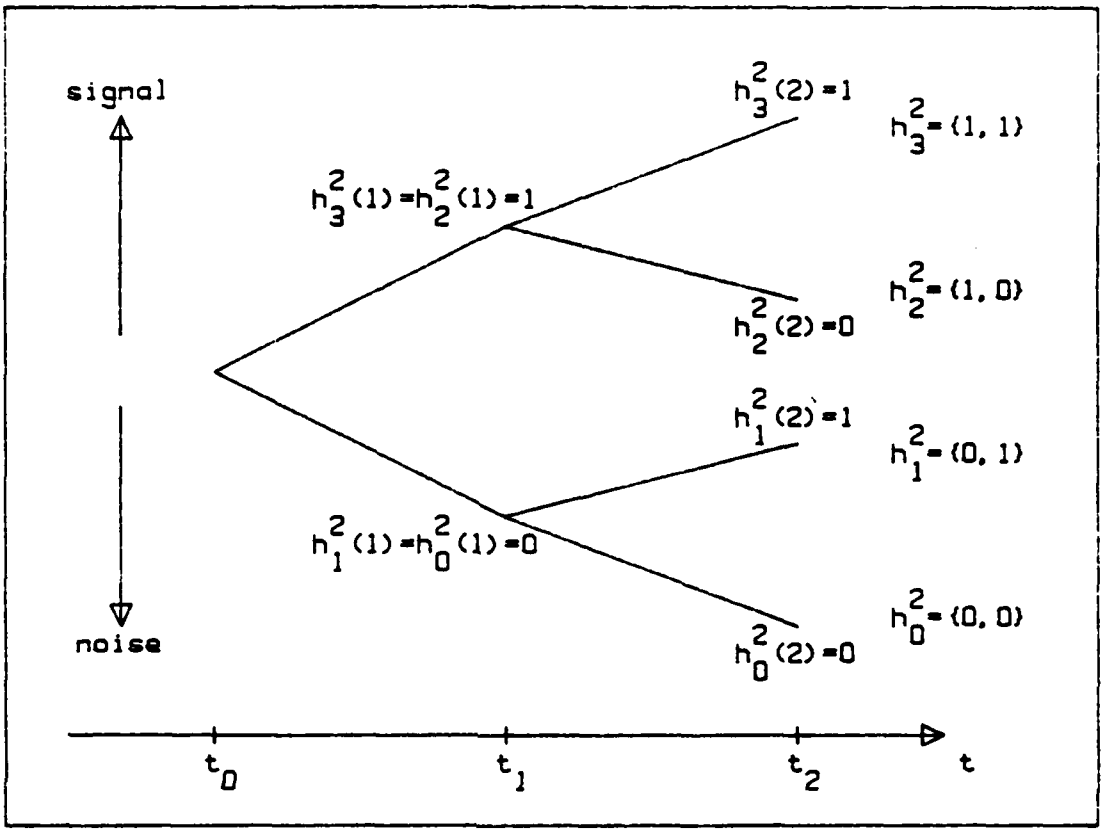


Figure 2. Hypothesis Sequences for Two Measurements.

The upward branches on the tree denote events attributed to the signal process while the downward branches are for those events assumed to be arising from noise.

With the addition of the second event at t_2, we observe in Figure 2 that each path of the tree splits into two separate hypotheses, reflecting the two possible choices for the source of the second event and giving rise to four possible hypothesis sequences. Looking back at equation (14), the four separate hypothesis sequences may be written as

$$h_0^2 = \{0,0\}$$

$$h_1^2 = \{0,1\}$$

$$h_2^2 = \{1,0\} \qquad (18)$$

$$h_3^2 = \{1,1\}$$

For this example, it can be seen that for any time interval, there are 2^{N_t} possible sequences. These hypothesis sequences are used as the models in the estimator developed by Meer. A Snyder and Fishman filter is associated with each hypothesis sequence, ignoring events postulated to be noise and incorporating those credited to the signal process according to the hypothesis sequence associated with that estimator.

A conditional probability of the correctness of each hypothesis sequence is also calculated based on the measurement sequence observed. These conditional probabilities are used as weighting functions to yield a single estimate of the mean

$$\hat{\bar{x}}(t) = \sum_{j=0}^{2^{N_t}-1} \Pr[h_j | Z^{N_t}] \hat{\bar{x}}_j(t) \qquad (19)$$

where $\Pr[h_j | Z^{N_t}]$ is the conditional probability of the correctness of hypothesis sequence j based upon a measurement history of N_t events, to be evaluated in the next section.

$\hat{\bar{x}}_j(t)$ is the estimate of the beam parameters generated by the subset of the measurement history associated with the jth hypothesis sequence.

The weighted sum of the covariances is calculated similarly:

$$\hat{\underline{\Sigma}}(t) = \sum_{j=0}^{2^{N_{t-1}}} Pr[h_j|Z^{N_t}]\{\hat{\underline{\Sigma}}_j(t) + [\hat{\bar{x}}_j(t)-\hat{\bar{x}}(t)][\hat{\bar{x}}_j(t)-\hat{\bar{x}}(t)]^T\} \quad (20)$$

where $\hat{\underline{\Sigma}}_j(t)$ is the filter covariance associated with the jth hypothesis. These formulae reflect a filter structure similar to that shown in Figure 3, where the measurements are incorporated into each elemental filter under its assoicated hypothesis and the estimates are then weighted and summed into an overall estimate. A major computational difficulty for this filter arises from the fact that the number of variables doubles with each additional event as the number of hypotheses needed increases, so some means of limiting the number of estimators required must be used.

Weighting Factors

The key to Multiple Model Adaptive Estimation lies in the use of the conditional probabilities as weighting functions. This section will examine how they are calculated.

The weighting factors are derived through the use of the signal and noise rate parameters evaluated in each hypothesis.

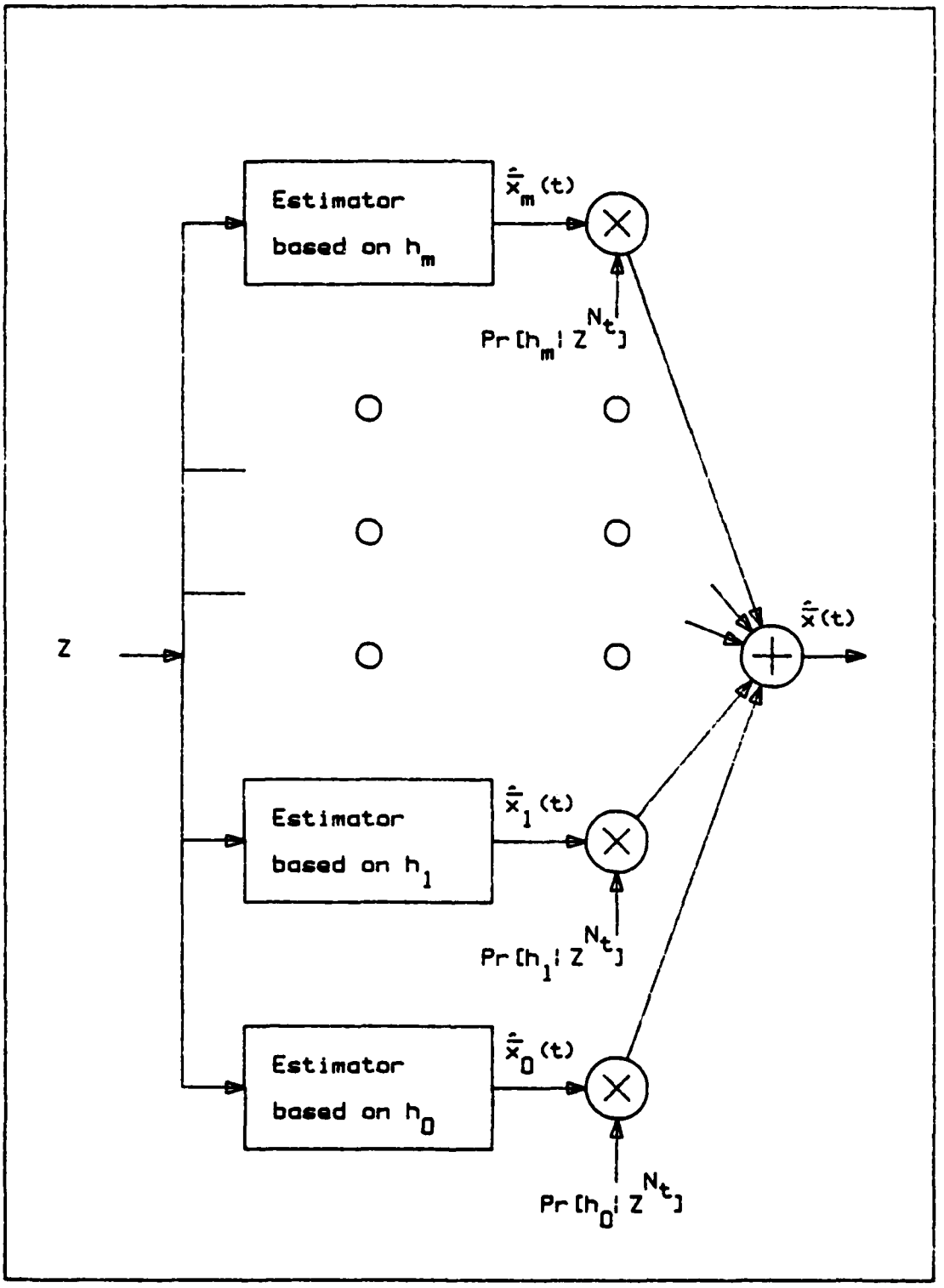


Figure 3. The Multiple Model Adaptive Estimator.

More precisely, the probability $\Pr[h_j^{N_t} | Z^{N_t}]$ that the jth hypothesis sequence is correct based upon a given measurement history can be computed with the equation

$$\Pr[h_j^{N_t} | Z^{N_t}] = \frac{\prod_{S_j} \hat{\lambda}_s(t_k, \bar{r}_k; \omega) \prod_{N_j} \hat{\lambda}_n(t_k, \bar{r}_k; \omega)}{\prod_{i=1}^{N_t} [\hat{\lambda}_s(t_i, \bar{r}_i; \omega) + \hat{\lambda}_n(t_i, \bar{r}_i; \omega)]} \tag{21}$$

where Z^{N_t} denotes the measurement history, and where the set of assumed signal indices under hypothesis j is defined as

$$S_j = \{k : h_j^{N_t}(k) = 1\}_j = \{k_1, k_1, \ldots, k_q\} \tag{22}$$

and the set of assumed noise indices under hypothesis j is defined as

$$N_j = \{\ell : h_j^{N_t}(\ell) = 0\} = \{\ell_1, \ell_2, \ldots, \ell_p\} \tag{23}$$

and $q+p=N_t$.

ω is a variable in the probability space Ω which specifies the random nature of the doubly stochastic space-time point process. A doubly stochastic space-time point process is a space-time point process in which some parameter of the process is itself random. Since $\bar{x}(t)$ is defined as random in equation (2) and we wish to allow other terms in $\lambda_s(\cdot)$ to be random, and to allow a random noise rate parameter as well, the particle beam problem may be readily described as

a doubly stochastic point process. One can actually observe a photon event at $(t,\bar{r})\varepsilon TxR^m$, where T and R^m are subspaces which span the possible n-dimensional realizations of the events, and thus gain knowledge of the inverse image, lying in a sample space Ω (Meer more rigorously defined a probability space consisting of a sample space, a Borel field, and a probability measure).

However, what we desire is to be able to distinguish between those events arising from the signal process and those arising from the noise process. To this end, we have postulated an unobservable probability space Ω_s which maps into the observable space Ω. This unobservable space is modeled as a cross product of three distinct probability spaces. The first of these, Ω_{s_1}, corresponds to the randomness of the signal process. The random nature of the noise process is represented by the probability space Ω_{s_2}. Finally Ω_{s_3} corresponds to the randomness of the hypothesis sequences. Thus it is possible to estimate the rate parameters by taking the expectation of these parameters over the appropriate probability space as follows

$$\hat{\lambda}_s(t,\bar{r};\omega) = E_{s_1}\{\lambda_s(t,\bar{r};\omega;\omega_{s_1})|Z^{N_t}\} \tag{24}$$

and

$$\hat{\lambda}_n(t,\bar{r};\omega) = E_{s_2}\{\lambda_n(t,\bar{r};\omega;\omega_{s_2})|Z^{N_t}\} \tag{25}$$

In Meer's simulation, equation (24) is simplified by evaluating equation (1) at the filter estimate, with Λ, $\underline{H}$ and $\underline{R}$ assumed to be time invariant to get the signal rate parameter for the measurement, i.e.

$$\hat{\lambda}_s = \Lambda \exp\{-[\bar{r}-\underline{H}\hat{\bar{x}}(t)]^T \underline{R}^{-1}[\bar{r}-\underline{H}\hat{\bar{x}}(t)]/2\} \qquad (26)$$

The evaluation of equation (25) is made even simpler by modeling the noise as uniformly distributed over the detectors' field of view. Thus $\hat{\lambda}_n$ is simply a constant for the filter model.

From the form of equation (21), it can be seen that the weighting factors are amenable to being calculated recursively. Thus it is only necessary to calculate the current values of the rate parameters based upon the latest event to update the weighting factors, rather than storing the whole event and estimation history.

Limitation of Hypotheses

The key to limiting the growth of the number of estimators needed lies in the use of the conditional probabilities already in use as the weighting functions. This section will examine two methods which may be used to limit the estimator's growth.

Best Half Method. We will begin with the method developed by Meer in his dissertation. The basis of this method is to decide which hypothesis for the oldest retained event

(the event which occurred at t_i where $i=N_t-D+1$, where D is the memory Depth) in the hypothesis tree is the most correct and to discard that half of the elemental filter bank corresponding to the less correct hypothesis. This effectively limits the number of elemental filters to 2^D.

The algorithm for the best half method initially begins the same as the MMAE filter with a growing hypothesis tree until the number of events observed equals D, the desired depth of the hypothesis tree. Once the number of events observed exceeds D, it becomes necessary to eliminate the less accurate half of the hypothesis tree. This decision is made by summing the weighting factors for the lower half of the tree to obtain an estimate of how correct that set of hypotheses is, then doing the same thing for the upper half of the hypothesis tree. By "lower tree half", we mean that portion of the tree corresponding to the most recent branches going downward (i.e., the most recent event is indicated to be a noise event), and similarly for the "upper half" corresponding to the most recent event being indicated as signal.

Now that we have a probability of how correct each half of the tree is, we compare them. If the hypotheses in the upper half of the tree are more probable than those in the lower half, we propagate the upper half of the estimates to the next event time and update them, while discarding the lower half of the tree. Also the weighting factors must be

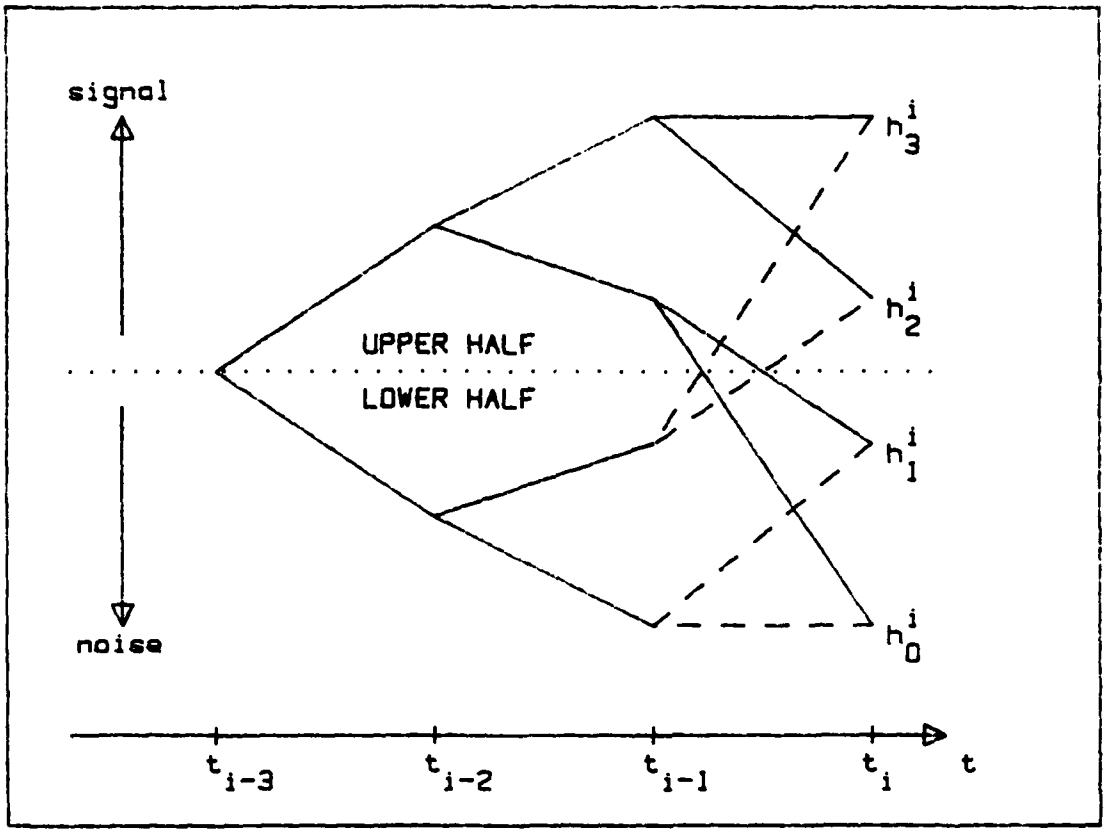


Figure 4. Best Half Method.

normalized so that the reduced set of weighting factors will still sum to unity. In the opposite case, the lower half of the tree is the one retained while the upper half is ignored.

Figure 4 may help in understanding the algorithm by illustrating it for the case of D=2. The solid lines depict the propagation and update paths when the upper half is the most probable set of hypotheses. The dashed lines depict the propagation and update paths when the lower half is the more probable set of hypotheses.

Merge Method. While the best half algorithm makes good

use of the possible measurement history, it is discarding a portion of the possible hypotheses, in order to limit the memory requirements of the estimator. Weiss and co-workers (Ref 36) have proposed another method for limiting the hypothesis tree that attempts to preserve that history more fully.

The key to the merge method lies in applying the principle of MMAE to pairs of hypotheses within the hypothesis tree. We begin by identifying two hypotheses which are identical except for the oldest event represented; thus we consider whether the oldest event in our hypotheses was a noise event or a signal event:

$$h_j^{N_t}(N_t-D+1) = \{0\}$$
$$h_k^{N_t}(N_t-D+1) = \{1\} \tag{27}$$

with all other events in the two hypotheses equal. In our numbering scheme $k=j+2^D/2$. A weighted sum of the estimates associated with these two hypotheses is taken using forms of equations (19) and (20) modified as follows

$$\hat{\bar{x}}_j'(t) = \{Pr[h_j|Z^{N_t}]\hat{\bar{x}}_j(t) + Pr[h_k|Z^{N_t}]\hat{\bar{x}}_k(t)\}/a \tag{28}$$

and

$$\hat{\Sigma}_j'(t) = (Pr[h_j|Z^{N_t}]\{\hat{\Sigma}_j(t)+[\hat{\bar{x}}_j'(t)-\hat{\bar{x}}_j(t)][\hat{\bar{x}}_j(t)-\hat{\bar{x}}_j'(t)]^T\}$$
$$+ Pr[h_k|Z^{N_t}]\{\Sigma_k(t)+[\hat{\bar{x}}_k(t)-\hat{\bar{x}}_j'(t)][\hat{\bar{x}}_k(t)-\hat{\bar{x}}_j'(t)]^T\})/a \tag{29}$$

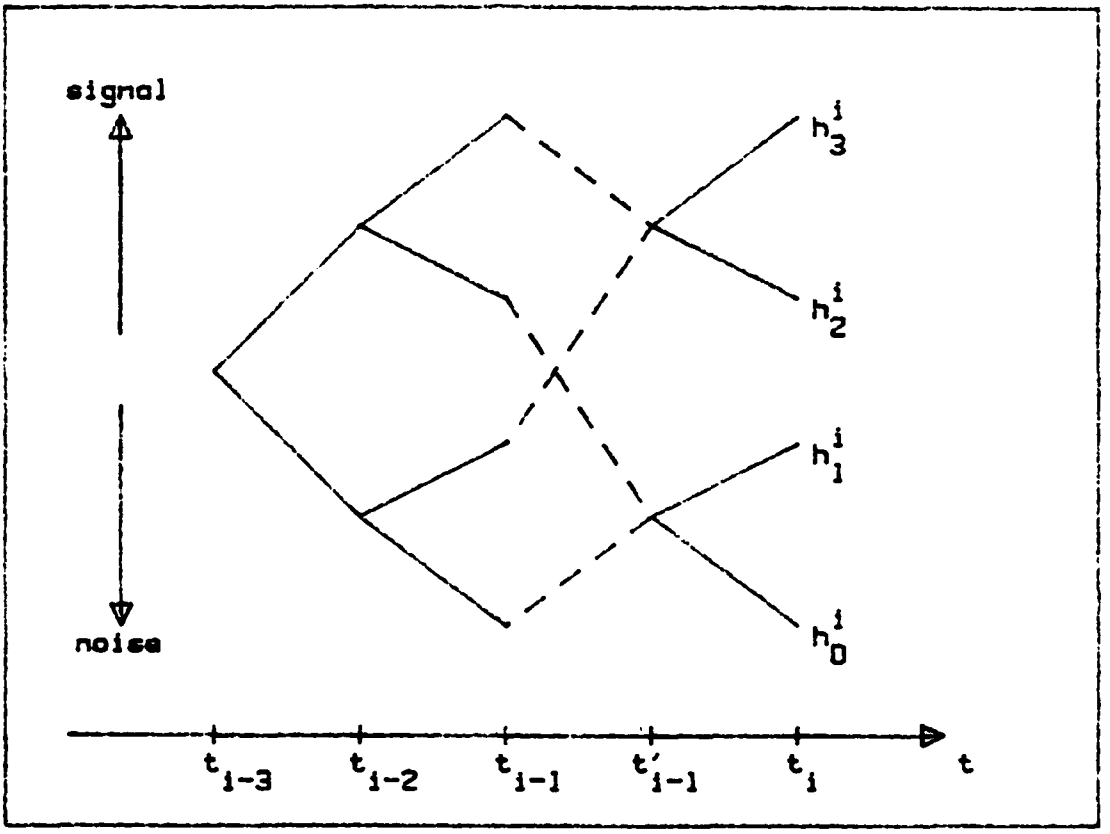


Figure 5. Merge Method.

where

$$a = Pr[h_j' | Z^{N_t}] = Pr[h_j | Z^{N_t}] + Pr[h_k | Z^{N_t}] \qquad (30)$$

These calculations are performed for j=0 to $j=2^{D-1}-1$. In Figure 5, the method's application for the D=2 case is illustrated. The solid lines depict the previously explained hypothesis growth. The dashed lines show how the intermediate step at t'_{i-1}, immediately after the elemental updates and the weighted summation are performed via equations (12), (13), (19), and (20) and before the hypothesis doubling can take

place, introduced to reduce the number of hypotheses by half to prevent growth beyond the imposed limits.

This method has the advantage of retaining in some measure the full set of possible hypotheses while maintaining limits on the number of elemental estimators used and is thus potentially more accurate than the best half algorithm. However, the method is also somewhat more difficult from a computational standpoint, making the choice of method used dependent upon the application.

Summary

In this chapter, we have outlined the Snyder and Fishman estimator which is used as the basic element of the MMAE. The hypothesis tree underlying the MMAE structure used by Meer was explained. Also shown was the calculation of the weighting factors. Finally, the Best Half and Merge methods for limiting the size of the estimator was outlined for the reader.

In the following chapter, the design of the full state feedback controller to be used with the Meer filter for the purposes of regulation and tracking with the beam will be developed.

III Controller Design

This chapter will go through the development of the controller implemented for both the Regulator and the Tracker problems. The first section deals with the design of the constant gain regulator used. The next section deals with the target model and the tracking filter associated with it for use in the tracker simulation. Finally, the constant gain tracker is developed.

The controller design is initially based upon the assumption of a Linear system, Quadratic cost criteria and full-state feedback. By assuming that the states are perfectly known, it is possible to derive an LQ regulator and tracker through the use of backward Riccati difference equations. Then, by assumed certainty equivalence one can provide the regulator the output of the Meer filter, and the outputs of a Meer and Kalman filter for the tracker, rather than the actual states, to synthesize an appropriate controller.

Regulator

Some further assumptions governing the implementation of the regulator controller were the use of a one-dimensional first-order system model, and the decision to use constant gains in the controller. This was done primarily to keep the computations simple and to limit the performance analysis requirements. For these assumptions, the cost-minimizing con-

troller takes the general vector form

$$\bar{u}(t_i) = -\underline{G}_c \hat{\bar{x}}(t_i) \tag{31}$$

where this term is added to the system differential equation (2) and the estimator propagation equation for the mean (10). Thus, under the first-forder and time invariance assumptions, these equations become, respectively,

$$\dot{\underset{\sim}{x}}(t) = (-1/\tau)\underset{\sim}{x}(t) + u(t_i) + G\underset{\sim}{w}(t) \tag{32}$$

$$\dot{\hat{x}}(t) = (-1/\tau)\hat{x}(t) + u(t_i) \tag{33}$$

where τ is the system's time constant.

The quadratic cost function we are attempting to minimize has the form

$$J = E\{\sum_{i=0}^{N} [\bar{x}^T(t_i)\underline{X}(t_i)\bar{x}(t_i) + \bar{u}^T(t_i)\underline{U}(t_i)\bar{u}(t_i)] + x^T(t_{N+1})\underline{X}_f\bar{x}(t_{N+1})\}/2 \tag{34}$$

where the $\underline{X}(t_i)$ and $\underline{X}_f$ weighting matrices are positive semi-definite and the $\underline{U}(t_i)$ matrices are positive definite.

The gains in (31) are generated from the solution of the backqard Riccati recursion

$$\underline{G}_c(t_i) = [\underline{U}(t_i)+\underline{B}_d^T)t_i)\underline{K}_c(t_{i+1})\underline{B}_d(t_i)]^{-1} [\underline{B}_d^T(t_i)\underline{K}_c(t_{i+1})\_(t_{i+1},t_i)] \tag{35}$$

$$\underline{K}_c(t_i) = \underline{X}(_i) + \underline{\Phi}^T(t_{i+1})\underline{K}_c(t_{i+1}) \cdot[\underline{\Phi}(t_{i+1},t_i)-\underline{B}_d(t_i)\underline{G}_c(t_i)] \quad (36)$$

solved backwards from the terminal condition

$$\underline{K}_c(t_{N+1}) = \underline{X}_f \quad (37)$$

For the purposes of this effort $\underline{B}_d(t_i)$, the control input matrix, is assumed to be unity, as is the weighting matrix $\underline{X}(t_i)$, because the X/U ratio is the important factor in the determination of the steady-state gains. The weighting matrix $\underline{U}(t_i)$ and the state transition matrix $\underline{\Phi}(t_{i+1},t_i)$ become scalar constants for the first-order system under consideration. Because we desire to implement a constant gain controller, all that is desired is the steady state solution to the Riccati recursion equations (35)-(37). So it is possible to simplify the recursion equations to the algebraic equations

$$K^2 + K[U-1-\Phi^2] - U = 0 \quad (38)$$

$$G_c = \Phi K/(U+K) \quad (39)$$

Because the equations used to derive the feedback gain were difference equations, the control value was assumed to be added into the equations at discrete intervals. But the measurement update times are distributed randomly, so it is necessary to distribute the control as a constant input to the dif-

ferential equations (32) and (33) over the entire control interval. An appropriate "pseudo-continuous" gain is derived by inverting the discretization process

$$G_c\hat{x}(t_i) = \int_{t_i}^{t_{i+1}} \phi(t_{i+1}-\tau)G_r\hat{x}(t_i)dt \qquad (40)$$

yielding as the gain to be used in regulating the beam

$$G_r = G_c/\{\tau[1-\exp(-\Delta t_c/\tau)]\} \qquad (41)$$

where Δt_c is the time interval between the feedback sampling of the states.

As an aid to evaluating the regulator performance, two control modes were provided in the simulation. Mode one artificially draws the system state directly from the simulation of the beam, while mode two uses the estimate made by the Meer estimator. This allows direct assessment of the impact of the filter on the closed-loop system.

Target and Track Filter

In order to evaluate the performance of the Meer filter in a tracking problem, it was necessary to develop a target simulation and to implement a tracking filter to use with it. It was decided to model the target position as the output of a first-order shaping filter driven by white Gaussian noise

$$\dot{\underset{\sim}{x}}_r(t) = (-1/T)\underset{\sim}{x}_r(t) + G\underset{\sim}{w}(t) \qquad (42)$$

where $x_r(t)$ is the random variable describing the position of the target's image on the detector plane and T is the target time constant. This target model was chosen rather than the more realistic first order acceleration model for the sake of the simplicity of the LQ synthesis. Simplicity has been chosen over realism in this case because we are only involved in a feasibility study and wish to use only the simplest of controllers to avoid more complex controller design issues. For example, if a first order acceleration process was used, there would be poles on the imaginary axis and we could not be assured of a solution to the Riccati equation.

By specifying the strength of the measurement noise as well as the target's time constant and the strength of its driving noise, it was possible to implement the square root form of the Kalman filter embedded in the SOFE software for the tracking filter (Refs 17, 24). Thus we can avoid the numerical problems of the standard Kalman filter.

Because the filter and the target model were so well matched, it was felt that some unmodeled effects should be present in the target. The author decided to add a sinusoidal driving function to equation (42) of the form

$$V \cdot [\sin \nu t] \tag{43}$$

Because we do not wish the target to stray too far from the central region of the detector array, for example between plus or minus two centimeters, we use the following relationship

between the steady-state magnitudes of the input and output sinusoids

$$2_{cm} = V/[\sqrt{(\nu^2 + T^{-2})}] \tag{44}$$

making it possible to define the following relationship

$$\nu = \sqrt{(V/2)^2 - T^{-2}} \tag{45}$$

Tracker

For the tracking problem, it is desired to minimize the error between the beam position and the target position. To use LQ controller synthesis, the two independent state processes are expressed as a single augmented state process, where

$$\bar{y}(t_i) = \underline{C}(t_i)\bar{x}(t_i) \tag{46}$$

$$\bar{y}_r(t_i) = \underline{C}_r(t_i)\bar{x}_r(t_i) \tag{47}$$

so that the error signal to be regulated to zero is expressed as (Ref 17:115, Vol III)

$$\bar{e}(t_i) = [\underline{C}(t_i) \quad -\underline{C}_r(t_i)] \begin{bmatrix} \bar{x}(t_i) \\ \\ \bar{x}_r(t_i) \end{bmatrix} \tag{48}$$

By extension of equation (31) and appropriate partitioning, it is possible to derive the feedback law (Ref 17:116, Vol III)

$$\bar{u}(t_i) = -\underline{G}_{c1}\bar{x}(t_i) - \underline{G}_{c2}\bar{x}_r(t_i) \tag{49}$$

Similar partitioning of the backward Riccati recursion equations reveals that the solution for $\underline{G}_{c1}$ is mathematically identical to that of the regulator problem presented in equations (35)-(37). The gain $\underline{G}_{c2}$ may thus be derived by the use of the additional equations

$$\underline{G}_{c2}(t_i) = [\underline{U}(t_i) + \underline{B}_d^T(t_i)\underline{K}_c(t_{i+1})\underline{B}_d(t_i)]^{-1} [\underline{B}_d^T(t_i)\underline{K}_r(t_{i+1})\underline{\Phi}_r(t_{i+1},t_i)] \tag{50}$$

where $\underline{\Phi}_r(t_{i+1},t_i)$ is the state transition matrix associated with the target model, and

$$\underline{K}_r(t_1) = -\underline{C}^T(t_i)\underline{Y}(t_i)\underline{C}_r(t_i) + \underline{\Phi}^T(t_{i+1},t_i)\underline{K}_r(t_{i+1})\underline{\Phi}_r(t_{i+1},t_i) - \underline{G}_{c1}^T(t_i)\underline{B}_d^T(t_i)\underline{K}_r(t_{i+1})\underline{\Phi}_r(t_{i+1},t_i) \tag{51}$$

$$\underline{K}_r(t_{N+1}) = \underline{C}^T(t_{N+1})\underline{Y}_f\underline{C}_r(t_{N+1}) \tag{52}$$

where for this problem the $\underline{C}$, $\underline{C}_r$ matrices, the weighting matrix $\underline{Y}$ and the control input matrix $\underline{B}_d$ are all set to unity. As we desire a constant gain for this portion of the controller as well, the steady-state solution of the equations simplify to

$$G_{c2} = [K_r \Phi_r / U + K_c] \tag{53}$$

$$K_r = -1/[1 + G_{c1}\Phi_r - \Phi\Phi_r] \tag{54}$$

substituting equation (54) into equation (53) yields

$$G_{c2} = - \Phi_r/\{[U+K_c \quad 1+G_{c1}\Phi_r-\Phi\Phi_r]\} \tag{55}$$

which, because of the random event distribution, must also be converted to a pseudo-continuous gain as in equation (41).

Four additional control modes were added for the tracker, to to help determine the impact of the Meer and Kalman filters on the closed loop system. These four modes correspond to all the combinations of artificial full-state knowledge versus feedback of estimated states for both the beam centroid state and the target location state.

Summary

In this chapter, the constant gain regulator and tracker were derived for controlling the particle beam. A target model was also determined for use in the simulation. The nest chapter will deal with the design of the simulation used to evaluate the filter and the controller.

IV Simulation Design and Performance Analysis

This chapter contains the simulation design used for the MMAE evaluation, and a discussion of the performance criteria which will be used in the next chapter. The first section will deal with the simulation software developed and used by Meer and the further modifications which were necessary to incorporate feedback control. The final section covers the evaluation of the performance parameters used to analyze the simulation results.

Simulation Design

For the simulation of his filter, Meer adapted SOFE (Simulation for Optimal Filter Evaluation), a software tool already in existence for the evaluation of fixed update interval extended Kalman filters (Ref 24). This was done by altering the executive so that one of the user-defined subroutines was called at random intervals governed by two inverse mappings of a Poisson distribution, to simulate the random nature of the event observation times. Into this routine he placed the algorithms necessary to propagate the elemental Snyder-Fishman estimators, to update the elements in accordance with their related hypotheses, to calculate the weighting factors and to generate the overall estimate as a probabilistically weighted average of the elemental

estimates. Also included was the algorithm for the Best Half method to limit the number of elemental filters required of the estimator.

This was done under the assumption that the beam centroid could be modeled as the output of a first-order shaping filter driven by white noise, thereby simplifying the elemental estimator equations for one spatial dimension to scalar equations. The weighting factors were calculated assuming R, the beam dispersion as well as Λ, the signal rate parameter constant, and $\hat{\lambda}_n$, the noise rate parmeter, to be time-invariant and perfectly known quantities. The rate parameters are defined by

$$\Lambda = \frac{\text{Expected number of signal events}}{\sqrt{2\pi R}[\text{duration of simulation}]} \tag{56}$$

$$\lambda_n = \frac{\Lambda\sqrt{2\pi R}}{L[SNR]} \tag{57}$$

where L is the length of the detector array and SNR denotes the signal-to-noise ratio. The system time constant τ, as defined in equation (32), and the square root of the strength of the driving noise $g=\sqrt{GQG^T}$, were also assumed to be perfectly known and time-invariant. The parameter values used in this study will be presented along with the discussion of the simulator results in Chapter V.

Discrete Feedback Update. In order to incorporate dis-

crete updates of the feedback control, it was necessary to separate the single routine that Meer had governing the propagation and update of the elemental filters into two routines, one dealing with the filters' propagation and the other updating the filters and calculating the new weighted estimate.

It was also necessary to add to the routine that determined the next event time for the estimators to be propagated to, an additional process which set up the fixed interval control feedback update times. This is because in the last chapter, it was decided to implement a fixed-sample-period controller, with the sampling period chosen on thebasis of the natural transients of the system and other sampling rate considerations, in the form of a feedback law involving the current best state estimates, whereas the filter's update periods are driven by the observed even times. Thus, the logic involved basically tests the current time of the subroutine call against each of the three processes (signal, noise or control) governing the next time to call the routine for each process, and determines if an event is to be incorporated or the feedback value updated. Then which ever process had given rise to the current call time calculates the next time that process will need to call the subroutine, with the feedback process based upon fixed time intervals, while the signal and noise processes are each generated by inverse mapping of a Poisson distribution from a random

number generator. Finally, the closest of these three future times to the current time is determined and the filter bank is propagated to that time. Figure 6 provides an illustration of the logic as an aid to understanding.

Inverse Poisson Mapping. Because the time intervals between the events of the signal and noise processes are described by a Poisson distribution, it was necessary to develop a method of generating random numbers that would fit the probability distribution function

$$f(\Delta t,\mu) = (1/\mu)\exp(-\Delta t/\mu) \tag{58}$$

where μ is the mean time between events for a process. By using a pseudorandom code to generate p, a random number uniformly distributed between zero and one, it was possible to obtain realizations of Δt which fit a Poisson distribution by inverse mapping equation (58) as follows

$$\Delta t = -\mu \ln(p) \tag{59}$$

This is what was implemented in Meer's simulation for each process to generate event times.

Inverse Gaussian Mapping. SOFE provides a generator for random numbers selected from a Gaussian distribution which operates by adding twelve random number calls from a pseudorandom code providing numbers uniformly distributed between zero and one. The Central Limit Theorem tells us

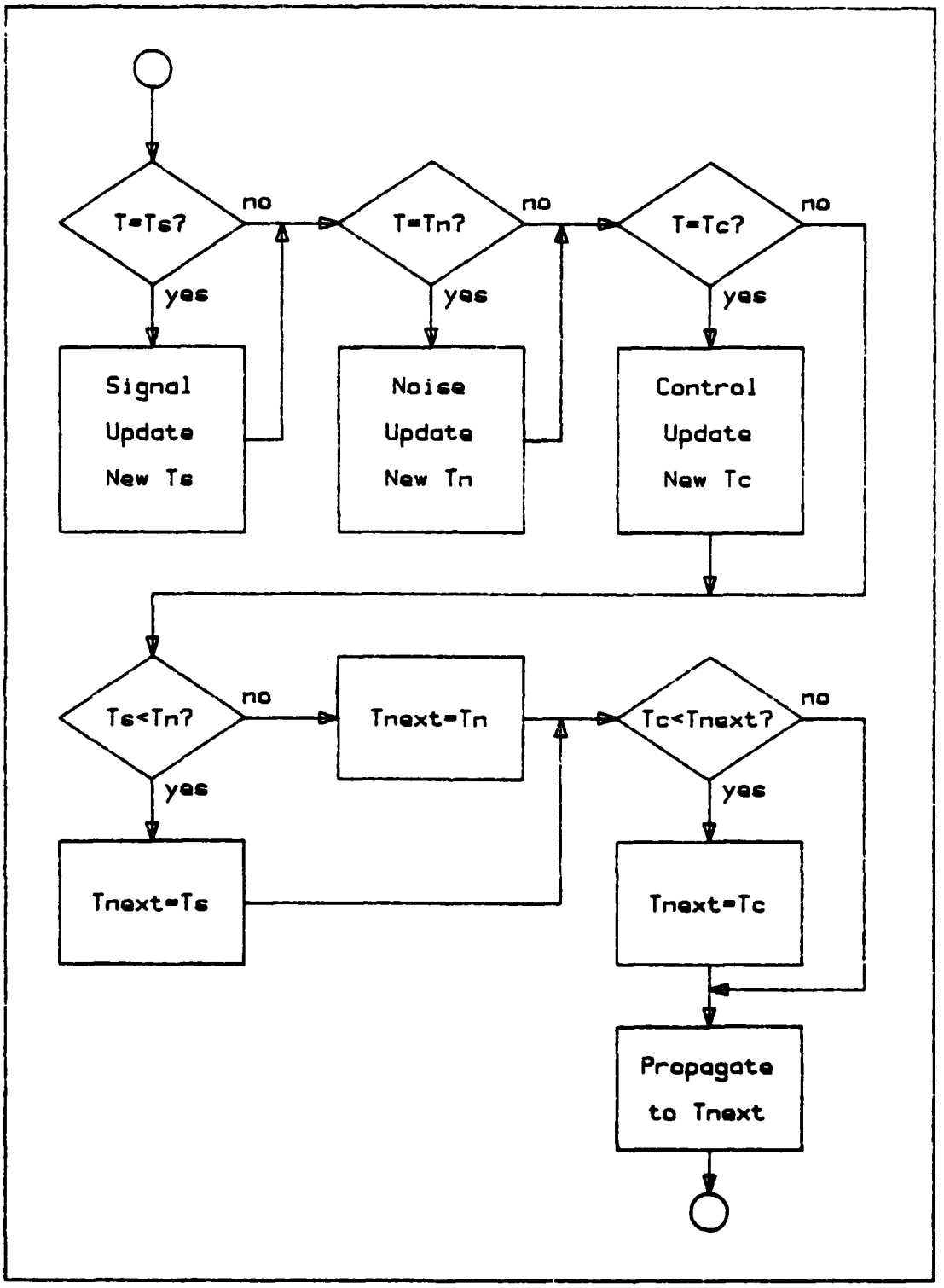


Figure 6. Event Simulation Logic.

that such a summation yields a close approximation to a Gaussian distribution with a mean of six and a standard deviation of one, so six is subtracted to get a zero mean and the result is then multiplied by the desired standard deviation and the mean of the desired distribution is then added. Meer did not alter this routine, but this author felt that, as the Poisson process generators used only a single random number call, it was conceptually more elegant and computationally more beneficial to substitute an inverse mapping of the Gaussian distribution.

$$\xi = \begin{cases} \mu + \sigma\sqrt{-\ln(|2p-1|)} & 2p \geq 1 \\ \mu - \sigma\sqrt{-\ln(|2p-1|)} & 2p < 1 \end{cases} \qquad (60)$$

where μ is the mean of the distribution, σ is the standard deviation, ξ is the realization generated, and p is again the output of the pseudorandom code used. Thus only a single random number call was used here as well. This also has the advantage of substantially reducing the number of calls to the pseudorandom code, which may be an important consideration because of the large number of Monte Carlo runs (200) required for adequate statistics in this case.

Performance Analyses

In this section a brief look will be taken at how the

performance parameters used to evaluate the filter performance and that of the Regulator and the Tracker are determined. This is done through the use of a software tool developed for use with SOFE known as SOFEPL, a Plotting Post processor for "SOFE", (Ref 25) which accepts the time history of all the Monte Carlo runs output by SOFE to produce sample statistics of means and covariances as functions of time.

The simulations are run over a period of 100 seconds. The results observed in Meer's dissertation were obtained by taking the value of the root mean squared (RMS) error at t=50 seconds. The RMS error is chosen as the measure of performance in this (and subsequent sensitivity tests) because it gives a measure of the error of the estimator from the true value regardless of sign. If we used the actual error, a positive error on one run could be canceled by a negative error on another run, resulting in an incorrectly low ensemble average. However, as the RMS error still exhibited some small variations with time, it was felt that a better performance parameter could be obtained by averaging the RMS errors observed for $t \geq 50$ seconds. The averaging of the RMS errors in the last half of the simulation is done to obtain a better estimate of steady-state error that is being converged upon by the Monte Carlo simulation. Because the time average of several values is taken, it is also possible to obtain a standard deviation about this time average that provides a

measure of how well the simulation is converging upon the steady-state error. We begin sampling the RMS error at t=50 seconds to minimize the effect of the filter startup transients on the performance results.

Because SOFEPL was designed primarily as an aid in filter design, its standard options derive ensemble statistics mainly for the error between a reference model state and a filter state, which are not suitable for judging the controller performance. However, SOFEPL does have provision for user-defined plots, by writing two small subroutines for insertion into the program. For the evaluation of the regulator, the first of these routines, which stores the sums necessary to perform the ensemble statistics, summed the simulations' beam state and the square of the beam state for each time that SOFE output it. The second subroutine performed the calculations necessary to get an RMS time history of the beam. This is an appropriate value to derive for judging the regulator performance since the regulator design was based upon a quadratic cost upon the closed-loop system's deviation from zero. If we were interested in doing more than a feasibility study for this control scheme, it would also have been appropriate to look at the RMS value of the applied control, but we are only examining the feasibility of the method, so these values were not considered. For the tracker, because the error to be minimized was that between the beam and the target,

the first subroutine stored the sums of that error and its square. It was not necessary to change the calculations in the second subroutine for the tracker to get the RMS error history of the simulation. These time histories were similarly averaged over the last half of the simulation period to avoid transients influencing the performance parameters.

Summary

In this chapter, we have shown the major features of the simulation to be used in evaluating the filter and controller performance. The parameters to be used in evaluating the performance have also been explained.

The next chapter will present the results gained from the simulation.

V Results

The results obtained from Monte Carlo simulations of the estimator and the controller are presented in this chapter. The goal of these simulations is to determine the relative effectiveness of the estimation and control methods presented in the earlier chapters as several of the major parameters are varied, and to investigate the sensitivity to these parameters. Although the actual values of the parameters used are appropriate for a tracking application, they are not taken from a specific problem. Therefore, the error performance results are useful as a relative measure of performance as the parameters are changed, rather than as an absolute measure of the estimators' performance in a specific application. In the first section, some preliminary results are presented, including sample plots from SOFEPL and the sensitivity of the simulation to the number of Monte Carlo runs used. In the next section, the two methods of limiting the estimators' size, Best Half and Merge, are compared while the sensitivity of the methods to six major parameters is shown.

In the third section, the performance of the regulator is presented with respect to the six parameters used previously. Finally, the sensitivity of the tracker to four parameters associated with the target is presented.

General Results

Figure 7 shows the true value of x and the multiple model adaptive estimator output averaged over 200 simulations of the filter. The true value of x is displayed by the solid line and the broken line is the filter's estimate of x. The values of the various parameters are listed on the figure. In all these figures, the expected number of signal point process events is 100 unless otherwise noted. The results displayed in this section are based upon the use of the Best Half method developed by Meer. These sample runs were made by passing to the filter the true initial conditions, x_0 = 5cm. The smoothness of $\hat{x}$ with respect to x is inherent because $\hat{x}$ is a weighted sum of eight elemental estimators, and the filter is given the exact dynamical model for x.

The ensemble average for the error statistics, for this 200 run example are shown in Figure 8. The solid line in Figure 8 is the ensemble average of the error, where the error is defined as $x-\hat{x}$. The two irregular broken lines are the ensemble averages of the error plus or minus the standard deviation of the error. The two relatively smooth dashed lines are plus and minus the average of the square root of the filter variance. The filter variance is the filter's estimate of how well it is performing. Of importance here is the fact that the filter's estimate of its error is similar to its actual performance and that the filter variance neither diverges nor goes to zero.

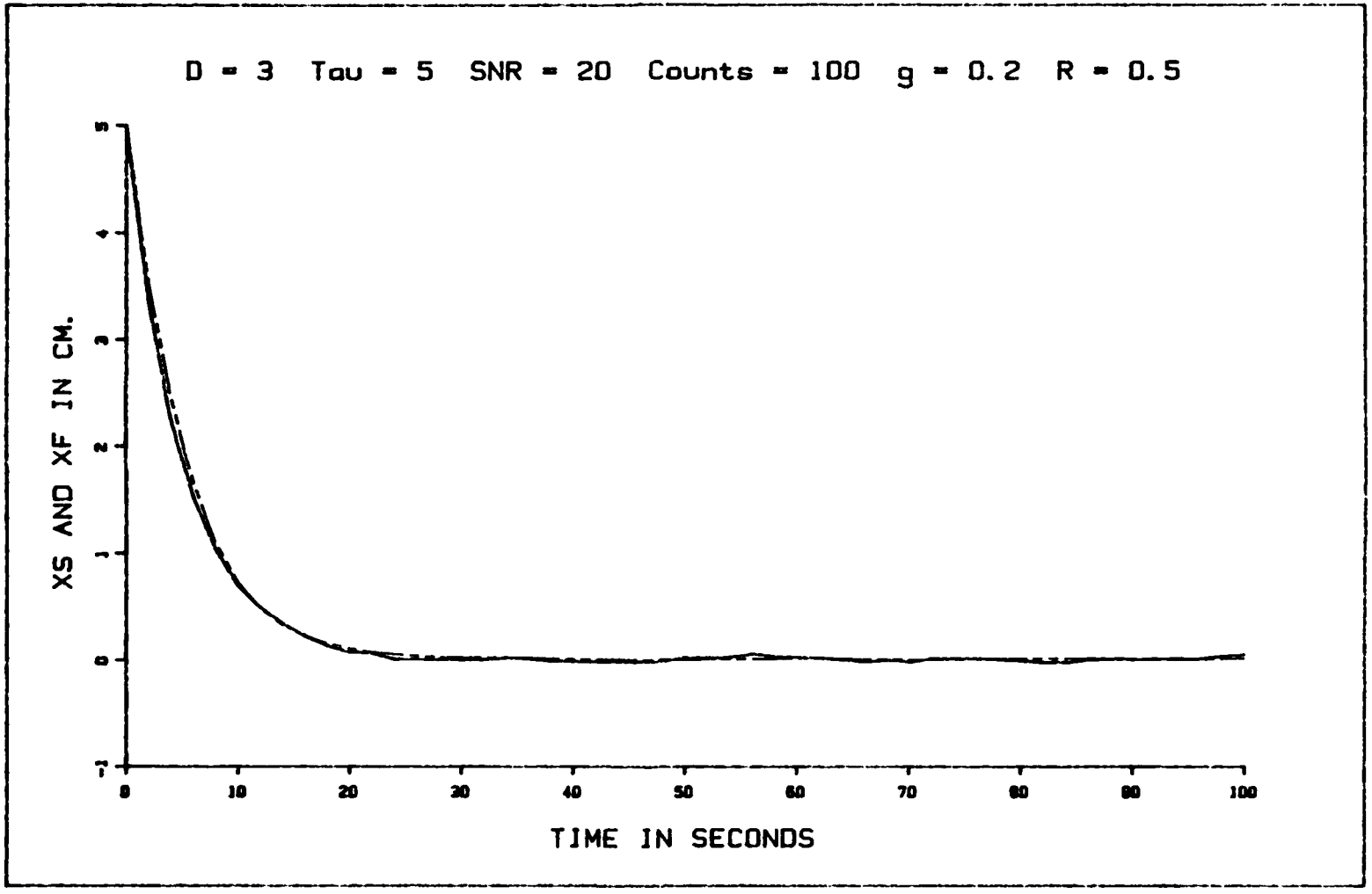


Figure 7. True and Estimated Ensemble Averages for 200 Runs.

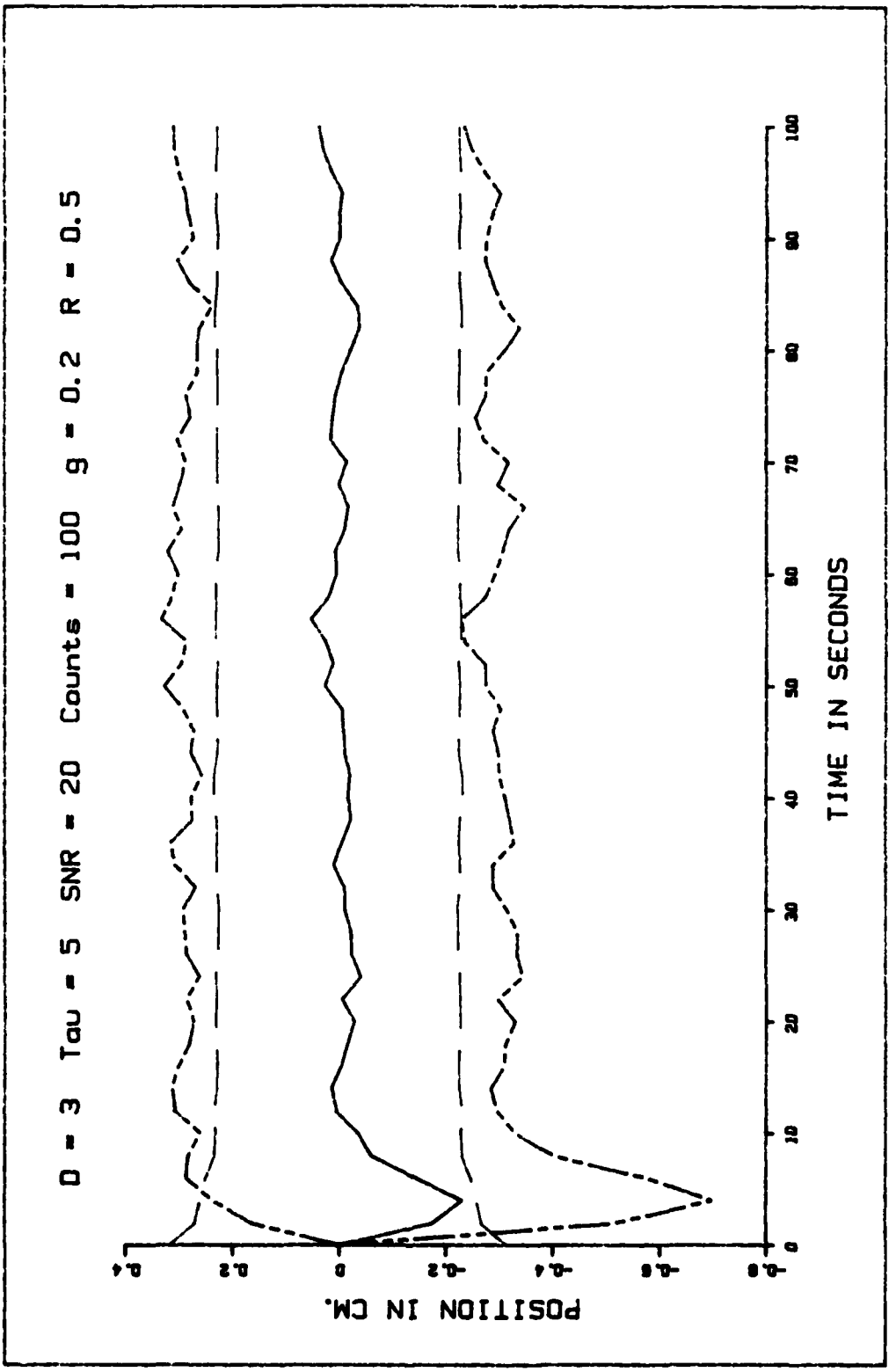


Figure 8. Ensemble Error Statistics for 200 Runs.

In Monte Carlo simulations, as more simulation runs are made the sample statistics become smoother. An important question is, how many runs are sufficient to give accurate performance data without expending an excessive amount of computation time? One method of addressing this question is to vary the number of simulation runs for a fixed set of input parameters and observe the error statistics. As more runs are made, the error statistics should converge to a final value. The results of this analysis are shown in Figure 9. In this figure, the number of runs is varied from two to 2000, and the average of the root mean squared (RMS) error for $t \geq 50$ seconds is plotted by the solid line. In Figure 9, we are not interested in the actual value of the RMS error. Instead, we are looking for a point beyond which there is little change in the error. The value of the error is erratic for less than 100 runs. For 100 or more runs, there seems to be very little convergence towards a steady state RMS error to be gained. Based on this, the plotted results in all the subsequent sections are for Monte Carlo simulations with ensemble averaging over 200 runs.

Best Half versus Merge

In the following sections, the sensitivity of the two formulations of the estimator to changes in several of their major parameters will be investigated. The general method is to vary one parameter while keeping the rest constant. In

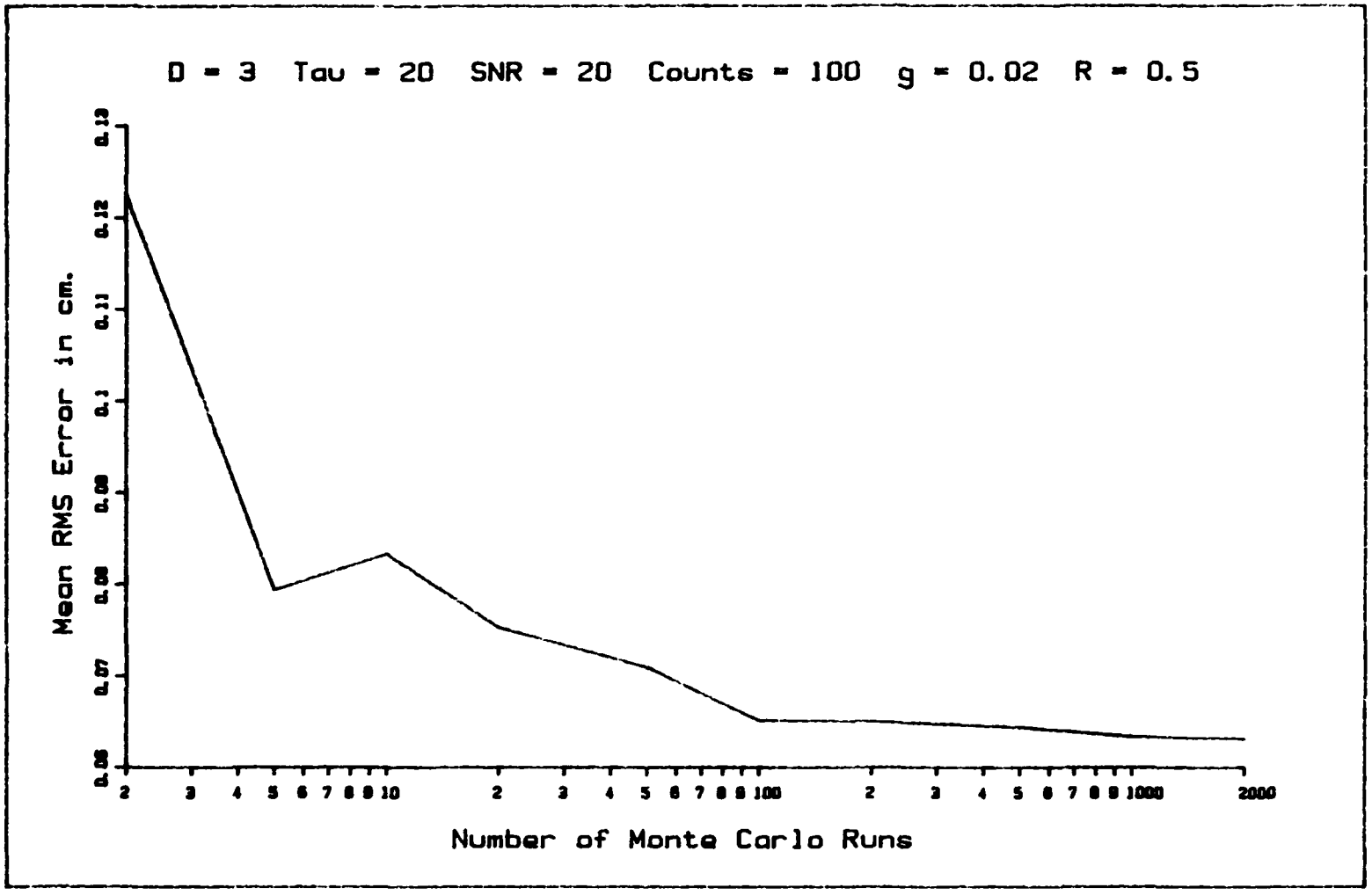


Figure 9. Sensitivity to Number of Monte Carlo Runs.

this study, six parameters associated with the processes and the estimator are examined; five of these parameters are varied about a design point. The first of these parameters is g, the square root of the driving noise strength, with a design value of 0.2 cm, and thus $\underset{\sim}{w}$ is in units of sec^{-1} to be compatible with Eq (2). Next comes the expected number of signal counts which is set to 100. The design point value of the time constant τ is 20 seconds. The signal to noise ratio is set to 20, a relatively high value. The depth of the estimator which determines the number of elemental estimators in use is 3; however, the depth sensitivity to be seen later is evaluated about a more demanding design point than the standard being discussed here. Finally, the beam dispersion used is 0.5 cm^2. The parameter values used are substantially the same as those used by Meer in evaluating the Best Half method in his simulation. This was done so that confidence could be established in the simulation results. The performance measure for the evaluation of the sensitivity is the time average of the RMS error for $t \geq 50$ seconds as discussed in the last chapter. In the figures accompanying the following sections, the solid lines with data points marked by B's depict the results obtained for the Best Half method, while the broken lines with data points marked by M's are the results obtained through the use of the Merge method. In all cases, the number of Monte Carlo runs is 200. We begin

by looking at the effect of the noise strength.

Sensitivity to g. The parameter g is the square root of the strength of the white Gaussian noise driving the dynamics of the unobserved process x, as seen in Eq (2). The curves in Figure 10 show the performance of the estimators as g is varied from 0.01 to 1.0 cm. The trend as expected; as the dynamic driving noise increases, the ability of the estimator to track the changes diminishes and the RMS error increases.

As can be seen from Figure 10, the RMS error for the two implementations is virtually identical for the lower noise levels. However, as the noise level rises above 0.1 cm, it can be seen that the Merge method provides slightly better results until the noise level increases to 1.0 cm where they again appear to be comparable. This phenomenon may arise from the fact that the driving noise g value is now larger than the beam dispersion used to calculate the weighting factors, so that the residuals from signal process events are now fairly large with respect to the beam dispersion, resulting in a substantial proportion of the signal events being rejected as if they were noise events. Because the weighting factors are used by the Merge method to maintain the limit on the estimator size as well as to determine the overall estimate, (as seen in Eqs (28)-(30) of the Merge section), it seems only natural for this error to have more impact on the Merge method.

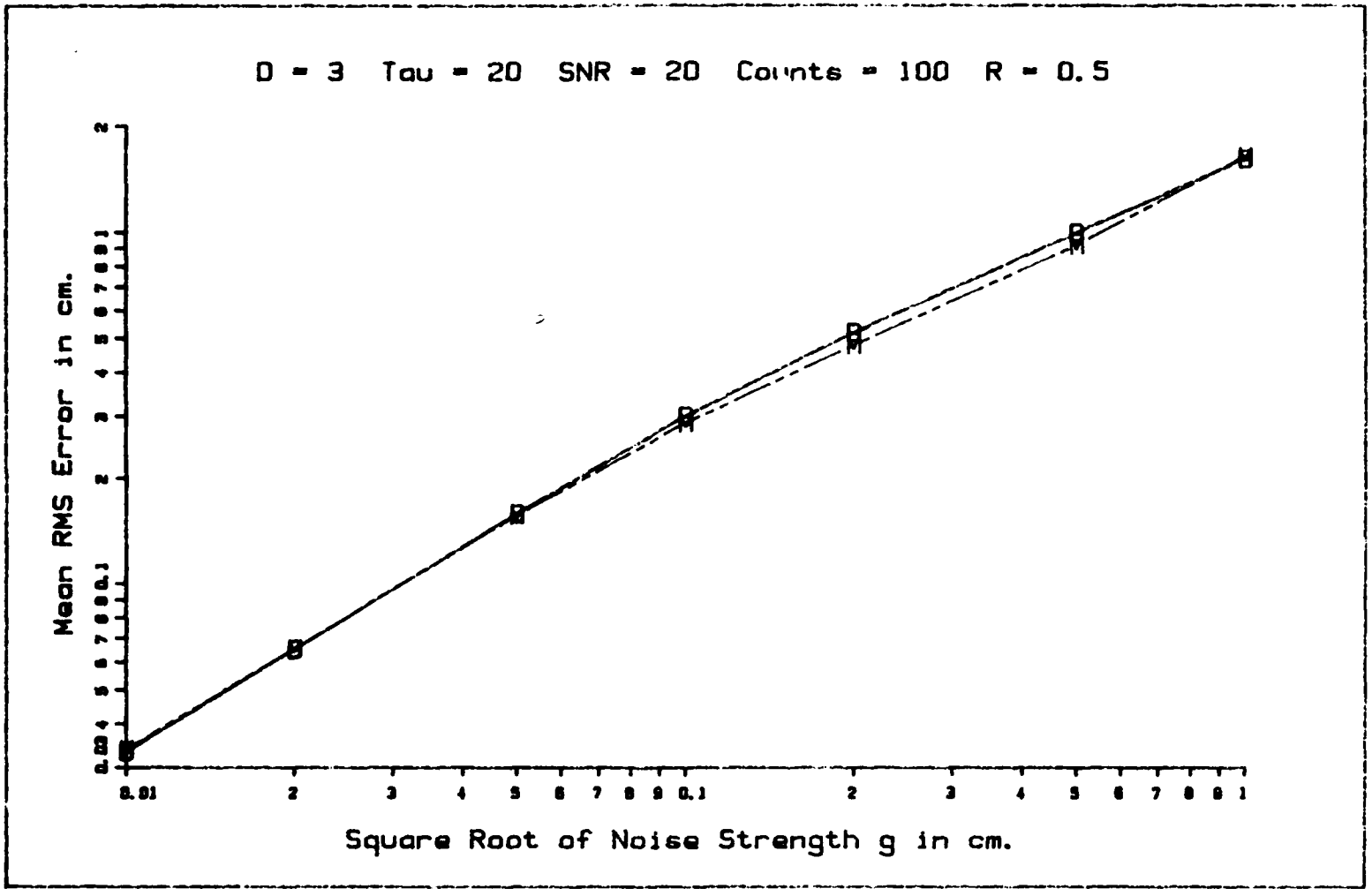


Figure 10. Estimator Performance Versus Dynamic Model Noise.

Sensitivity to Expected Number of Signal Events. In Figure 11, the time-averaged RMS error is plotted against the expected number of point process signal events. As might be expected, as more information is available to the estimator from the signal process, the RMS error goes down. It should be noted that, although the Merge method produces better estimates at the higher count rates than the Best Half method, it is also computationally more difficult and thus may be infeasible to implement at the highest count rates. Of particular interest is the more rapid deterioration of the Merge method for the lower count rates. It should be noted that the lowest data point graphed corresponds to an average time between updates of 10 seconds, which for the 20 second time constant used for beam dynamics is the Shannon sampling theorem (i.e. Nyquist rate) limit of the system. Because of the Merge method limiting the estimator size preserves the hypothesis history by taking a weighted sum of pairs of estimators, of which one member of the pair incorporates one less update than the other, the situation may be viewable as providing a slightly less effective updating of the covariance calculation than that provided by the Best Half method. This effect causes the Merge method to behave more poorly than the Best Half method near the sampling limit (further discussion of the gain weighting interpretation takes place in the section on weighting factor effects). By examining the relatively smooth

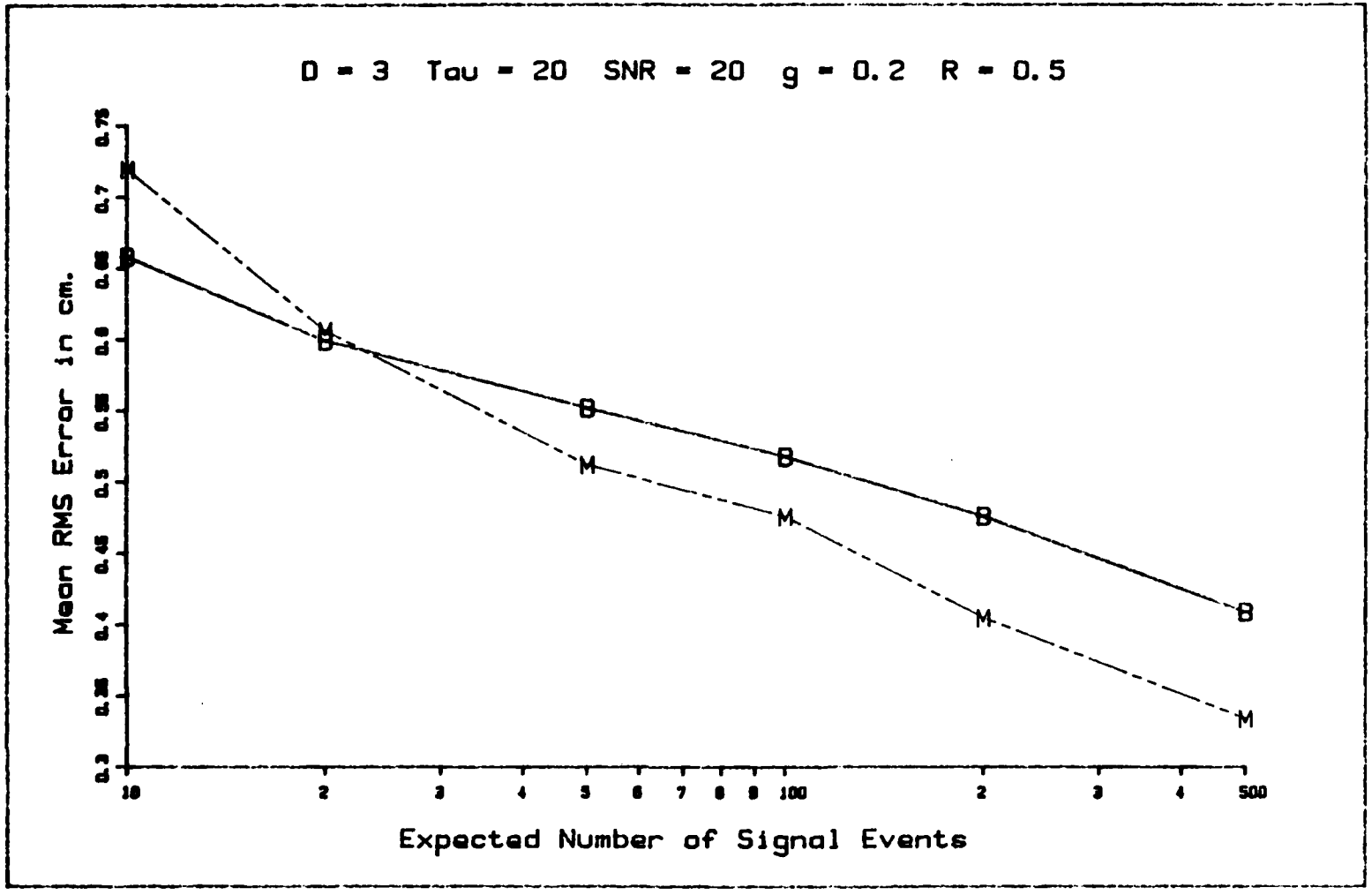


Figure 11. Estimator Performance Versus Expected Number of Signal Events.

Best Half method curve, it may be seen that the curve is also slightly steeper as it approaches the sampling limit. Thus the Merge method is merely deteriorating sooner because it reaches its effective update limit sooner than the Best Half method. As described before, we have assumed perfect knowledge of the model. It is expected that the RMS error would be much larger at the low count rates if the dynamical models of the true process and the filter were mismatched.

Sensitivity to τ. The sensitivity of the RMS error to τ is depicted in Figure 12. As can be seen, there is a strong trend toward increased RMS error as τ becomes larger. When τ is large, the dynamics of x depend proportionately more on the driving noise source and there is less restoring action due to the $\underline{F}(t)\bar{x}(t)dt$ term in Eq (2). This allows errors between x and $\hat{x}$ to persist for longer periods between signal induced point process observations. When τ is small, any errors caused by the driving noise in Eq (2) are rapidly reduced as the output decays to the steady state value.

As may be seen from the curves, the Merge Method does a better job for the longer time constants, but is indistinguishable from the Best Half method for the shorter time constants. This is as expected since the update rate remained constant for these curves. As the time constant gets smaller, the simulation is operating closer to the effective sampling rate limit observed earlier. If this parameter variation had been done at a higher sampling rate, the curves could be expected

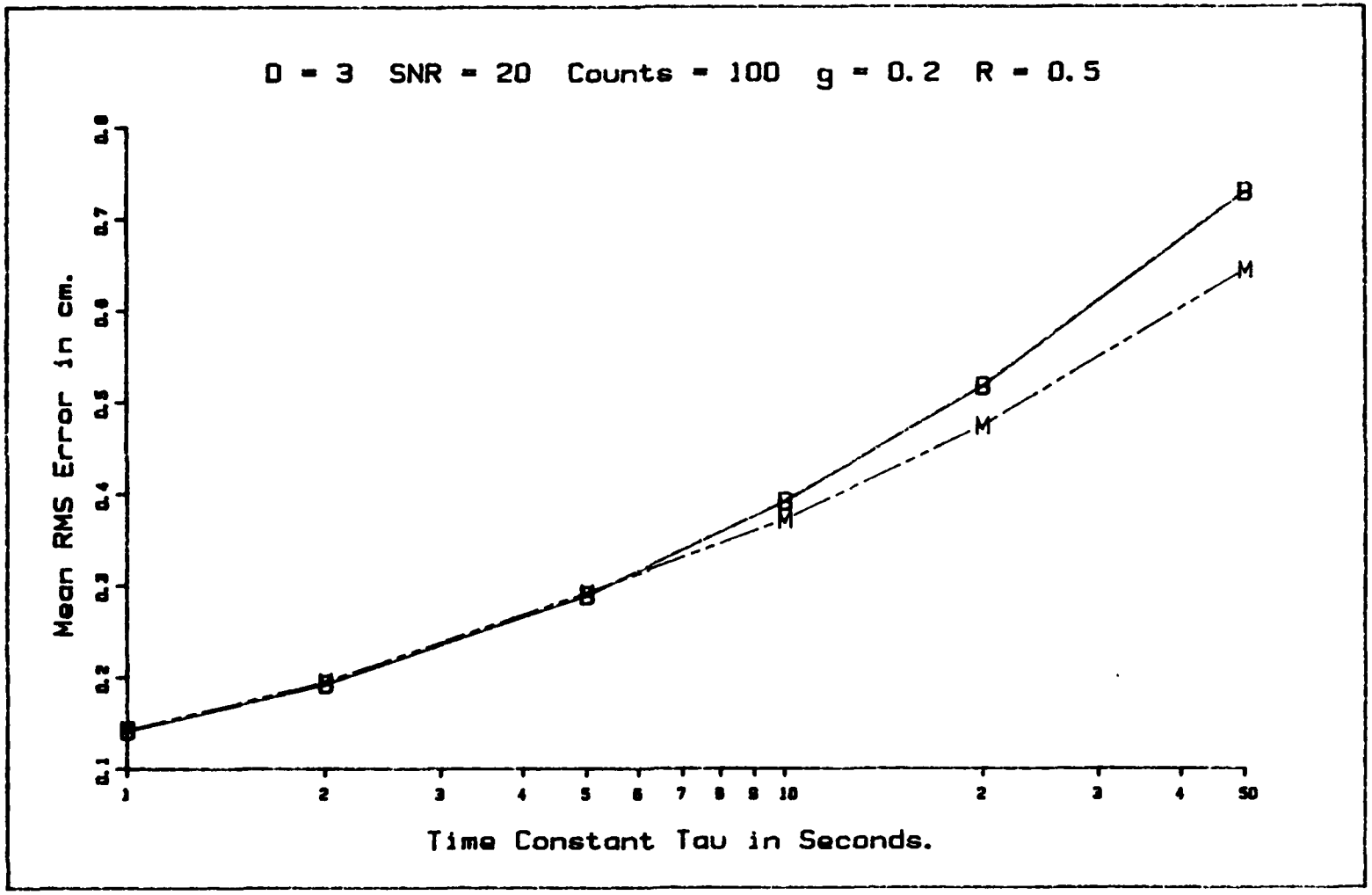


Figure 12. Estimator Performance Versus Tau.

to start diverging at a smaller value of the time constant. Also, as τ goes to 0, the process itself is becoming more white and any estimate is less useful, so the distinction between different forms of estimators becomes smaller.

Sensitivity to SNR. The results of the SNR sensitivity tests for both methods are shown in Figure 13. Note that, though both curves show a downward trend over most of the range of SNR, for values above 5, the curves seem to approach a limit asymptotically for their performance. The Merge method is seen to perform consistently better than the Best Half method, though there is a slight indication that the Best Half method is approaching another performance limit at the lower end of the SNR values used. However, it can be seen that the estimators' performance varies relatively little over a wide range of SNR and that the worst RMS error observed is only 0.68 cm for an expected signal to noise count ratio of 0.1. That is, one signal event is expected for every 10 noise events. It is expected that the RMS error would increase more rapidly if the true and filter models were mismatched.

Sensitivity to D. The sensitivity to depth, D, is shown in Figure 14. The design point parameters used in this section are different from those used in the other sections, with a large beam dispersion and driving noise and a smaller time constant and signal to noise ratio. This design point was chosen by Meer to stress the estimator's capabilities. Note

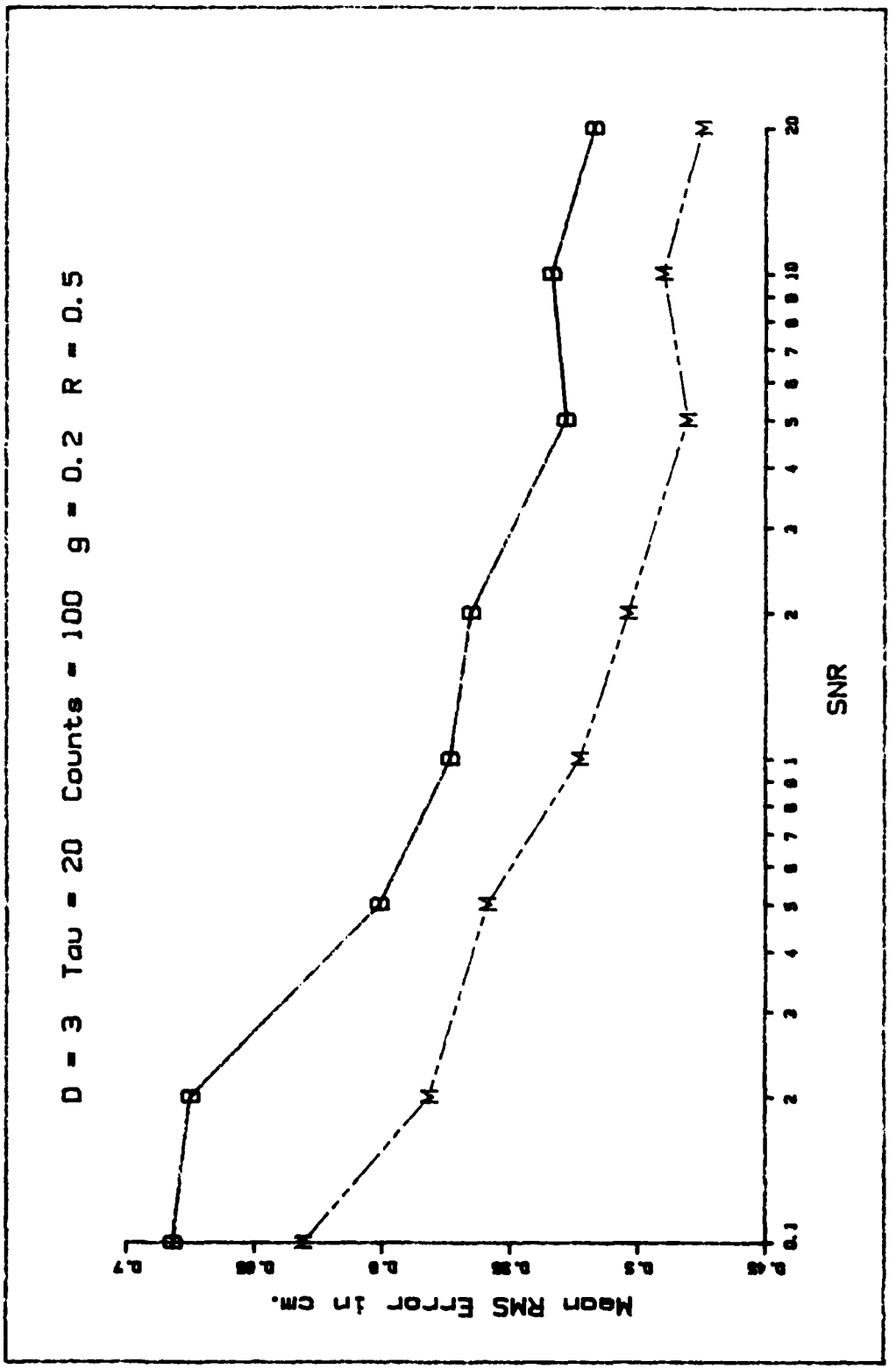


Figure 13. Estimator Performance Versus Signal to Noise Count Ratio.

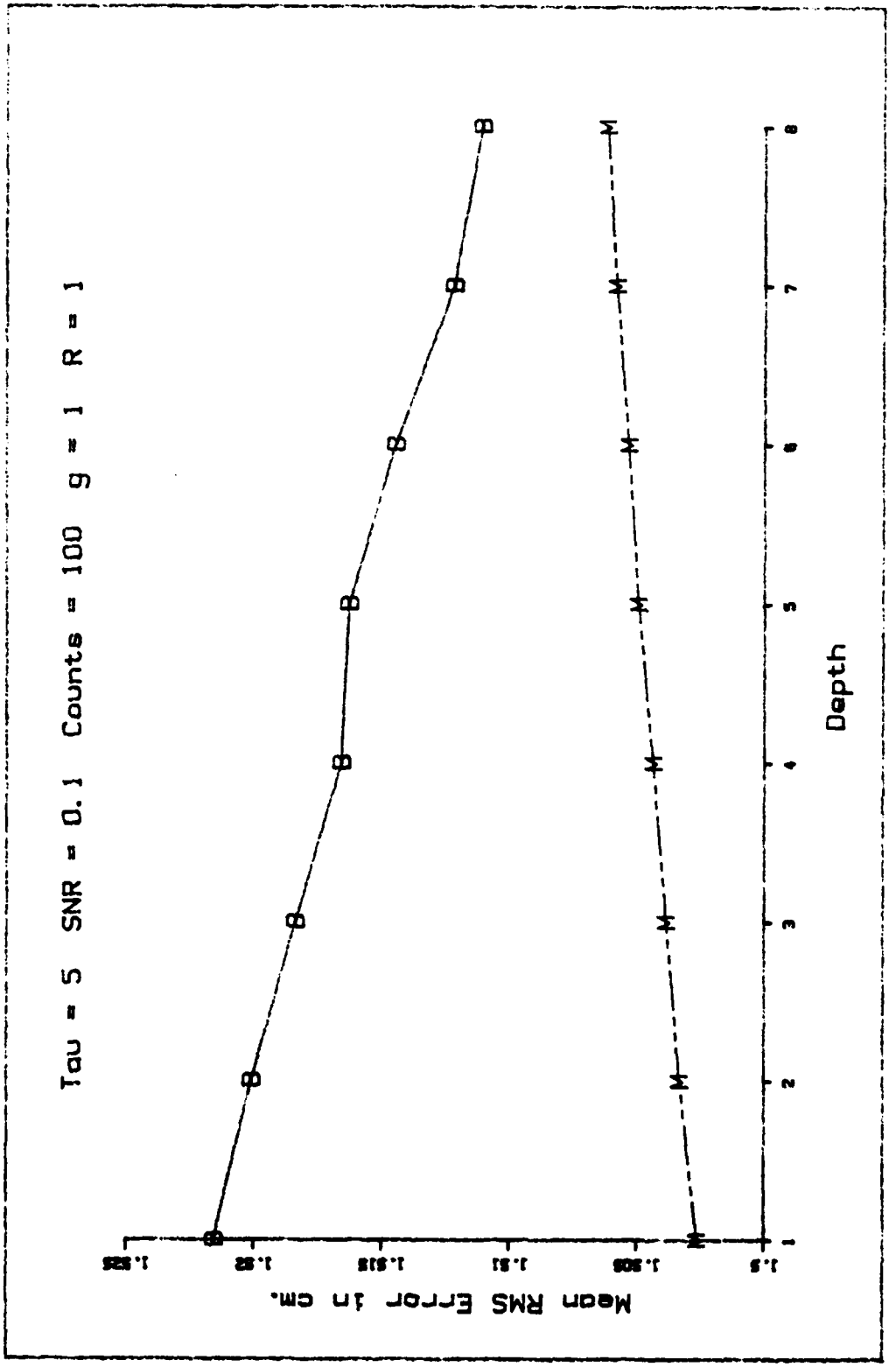


Figure 14. Estimator Performance Versus Depth.

that the error axis scale on this plot is greatly expanded and the variation of RMS error over a depth range of one to eight is very small. Because of this expansion, the author is hesitant to attach any significance to the fact that the slopes of the two curves have different signs. This suggests that at least if the model of the underlying process is well known, it may be possible to get by with only two elemental estimators and the weighting factor calculation to obtain acceptable performance. This is most likely a function of a scalar Gauss-Markov process x to define actual beam position. Higher order models will proabably make larger depths more beneficial.

Sensitivity to R. The sensitivity to R, the beam dispersion parameter, is displayed in Figure 15. It is apparent from the trend of the curves that as R becomes larger, noise induced events near the signal source are not deweighted as strongly as when R is small. Physically, this makes sense because as the beam dispersion becomes larger, the projection of the beam upon the sensor array takes up a larger proportion of the total array area. Thus the proportion of noise-induced events which fall within this area to those rejected because they are outside this area increases, thereby increasing the error. As can be seen, the Merge method performs consistently better, as might be expected since by performing the weighted summation of the covariances in Eq (29), the Merge method filter-computed covariance tends to be slightly higher than the Best Half meth-

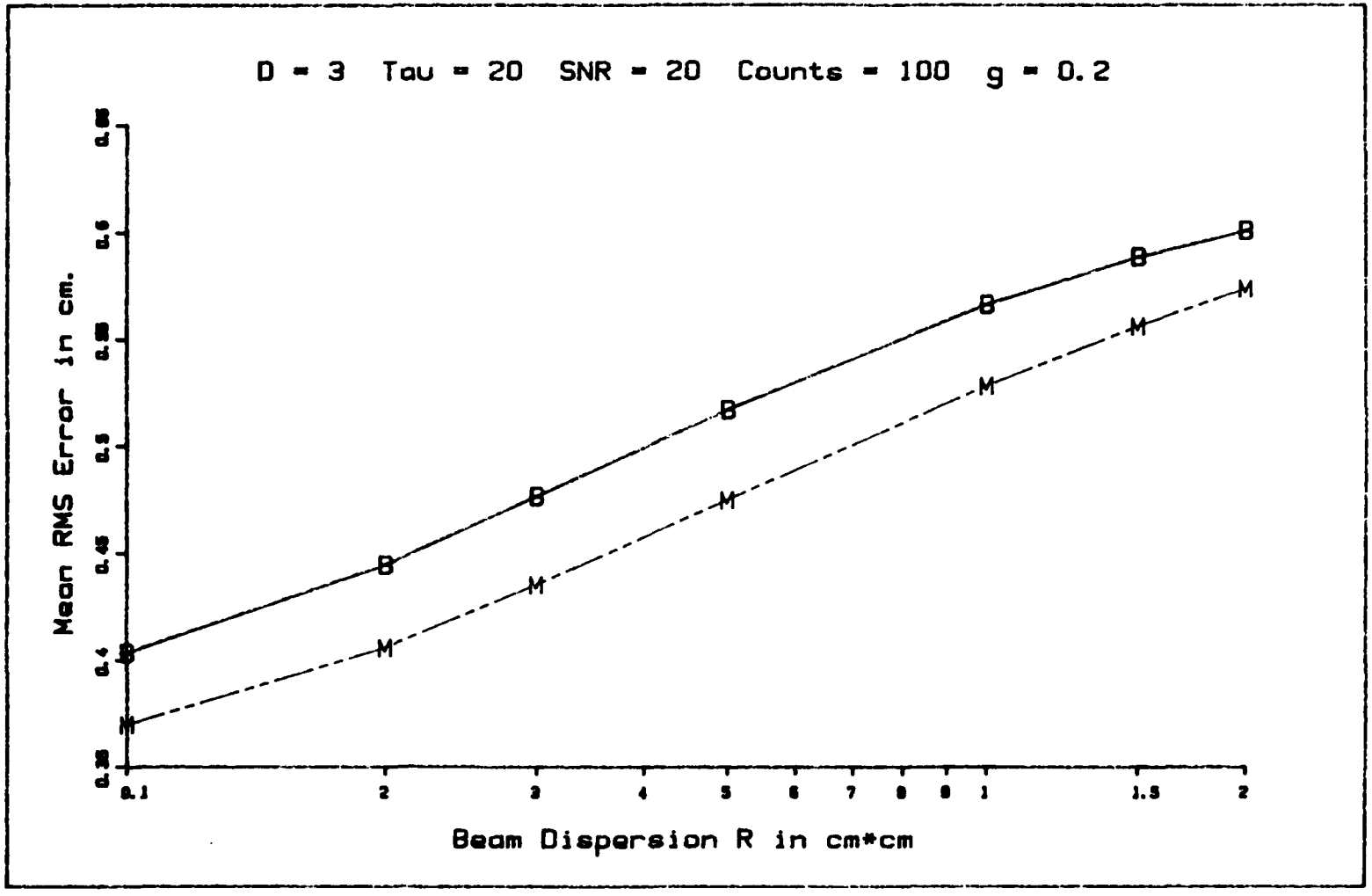


Figure 15. Estimator Performance Versus Beam Dispersion.

od's. This causes a higher gain to be applied to those events judged to be signal-induced, thereby compensating slightly for those signal-induced events erroneously judged by the estimator to be noise-induced.

Weighting Factor Effects

In this section, we are concerned with two issues arising from the use of the weighting functions, acquisition of the beam and the optimum matching of the observed statistics with the filter's predicted performance through its internally computed statistics.

Acquisition. Beginning with the acquisition issue, Figure 16 presents the ensemble averaged time histories from two simlation runs with identical parameters, using the Best Half and Merge methods. The upper two curves representing the true beam position for the two cases are not identical because different random number samples were used in the simulations. The important thing about these runs is that they used poor initial conditions: while the beam simulations begin at 5 cm, the estimator was started at 0 cm, the initial value of Σ was set at 0.1 cm^2 for the runs discussed, which is approximately twice the steady state value reached by the simulations. Thus, the actual initial condition was approximately 15 times the standard deviation the filter used for its initial conditions, which is a very taxing situation. The time history of the Best Half method estimator is depicted by the lowest dashed line in the figure. While the Merge method, depicted by the

broken line, yielded significantly better results, neither method can be said to have successfully recovered from the poor initial estimates. The better performance of the Merge method is as anticipated, since it attempts to retain in some measure those events which it has incorrectly judged to be arising from the noise. Another consideration which is not so easily seen in Figure 16 is that, while the Merge method estimate begins to appear to either side of the beam centroid position as early as t=50 seconds, the Best Half method is still slightly, though consistently, underestimating the position of the beam centroid as late as t=80 seconds. The most obvious method for correcting this problem is to increase the initial value of Σ used to reflect the range of possible error in the initial conditions. However, Meer (Ref 22:199) has shown that for small values of R this does not significantly help the estimator's acquisition performance.

A better understanding of the poor acquisition performance of the estimator may be gained by rearranging the update and merging equations for the D=1 case so that the two elemental estimators become a single estimator with the update gain multiplied by the weighting factor.

$$W(r-\hat{x}) = \hat{\lambda}_s(r-\hat{x})/(\hat{\lambda}_s(r-\hat{x})+\hat{\lambda}_n) \qquad (61)$$

where Eq (26) simplifies to

$$\hat{\lambda}_s(r-\hat{x}) = \Lambda \exp\{-(r-\hat{x})2/(2R)\} \qquad (62)$$

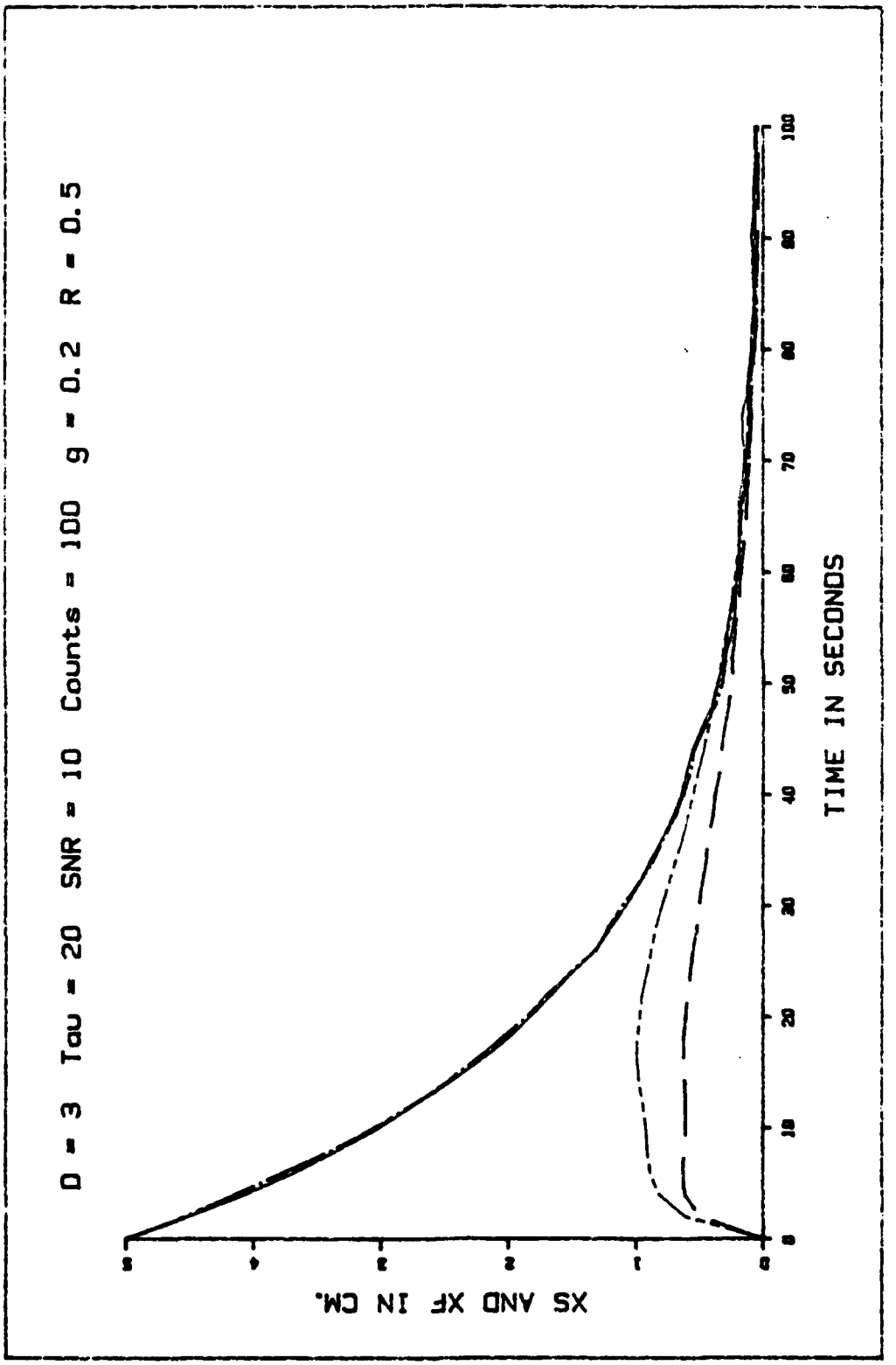


Figure 16. Unsuccessful Acquisition.

with Λ defined as in Eq (56), and where the signal-to-noise count ratio is defined as

$$SNR = \frac{\int_{-L}^{L} \hat{\lambda}_s dr}{\int_{-L}^{L} \lambda_n dr} = \frac{\Lambda\sqrt{2\pi R}}{\lambda_n L} \tag{63}$$

allowing the definition of the estimated noise rate parameter as

$$\hat{\lambda}_n = \{\Lambda\sqrt{2\pi R}\}/\{SNR \cdot L\} \tag{64}$$

This is illustrated in Figure 17, where the horizontal upper line shows the gain distribution used in an elemental Snyder and Fishman Estimator, while the curve below it corresponds to the effective gain distribution applied in the overall estimator as a function of the residual value. From the shape of this curve, it can be readily seen how events occurring far from the estimate can have very little impact upon that estimate. One method of correcting this problem without losing too much of the estimator's ability to distinguish between signal and noise process events is suggested by the fact that both of the estimator curves in Figure 16 rise very quickly to relatively constant values. It is proposed that instead of using the residual from the overall estimate to weight the signal versus noise weighting for all the elemental estimators as in Eq (26), to use the elemental filters' residuals to calculate elemental signal versus noise weightings.

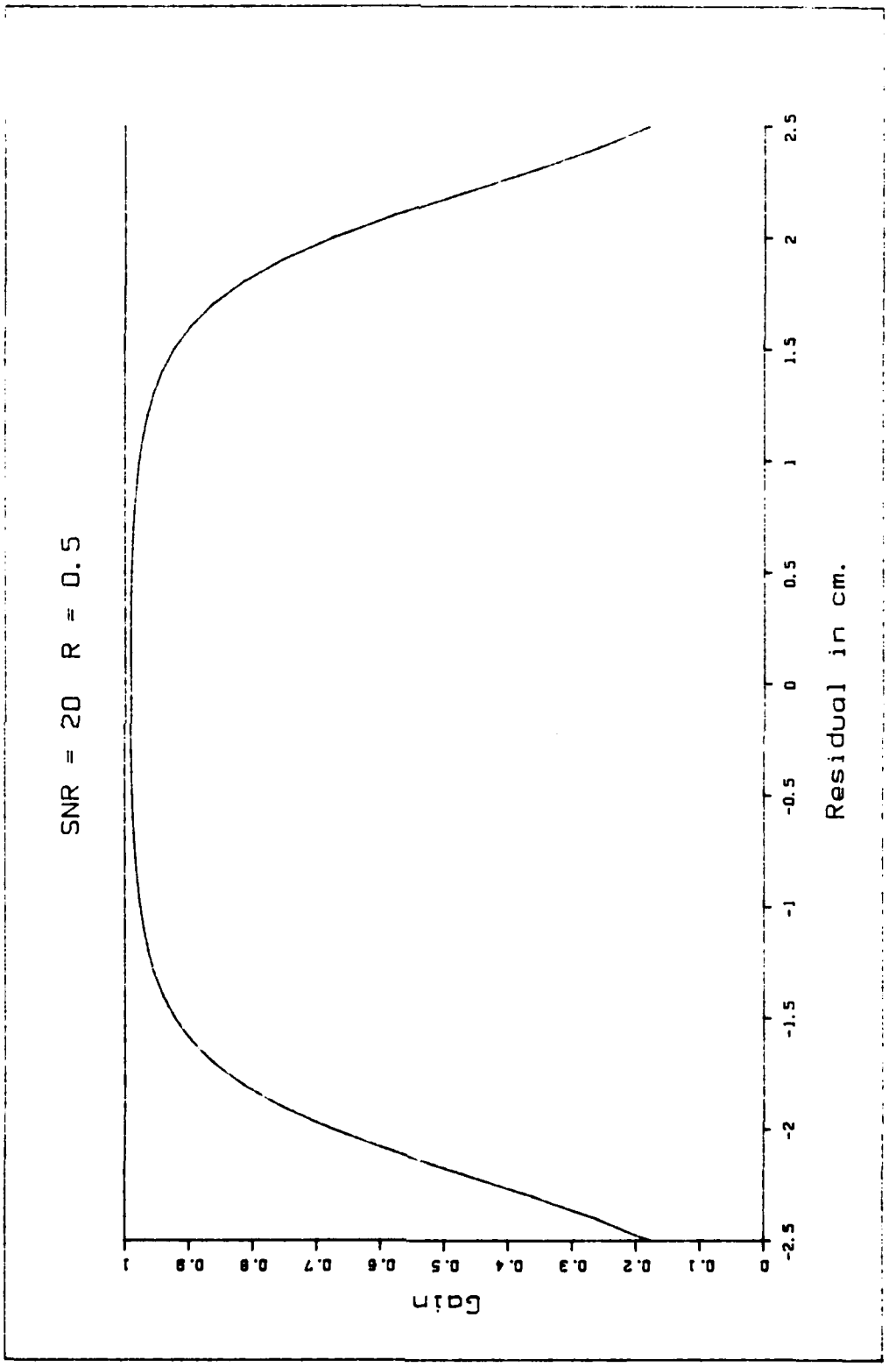


Figure 17. Gain Applied Versus Residual Value.

This causes Eq (26) to take the form

$$\hat{\lambda}_{s_j} = \Lambda \exp\{-[\bar{r}-\underline{H}\hat{\bar{x}}_j(t)]^T \underline{R}^{-1}[\bar{r}-\underline{H}\hat{\bar{x}}_j(t)]/2\} \qquad (65)$$

Thus those hypotheses which are incorporating the correct sequences of events will be given correspondingly higher weighting factors as they converge upon the beam centroid. This proposal has not been evaluated because it occurred to the author too late in the study to implement the software changes necessary.

Tuning. By comparing the RMS error observed in the simulations to the square root of the estimator's calculated error variance, it can be seen that the weighting functions give rise to another phenomenon. In Figure 18, the RMS error divided by the square root of the filter computed error variance is averaged over time as was done previously for the RMS error, and plotted versus a variation in τ. From this figure it can be seen that, for the longer time constants, the observed RMS error is significantly greater than the predicted variance of the filter would suggest. The cause of this effect may be discerned by examining the effective gain curve in Figure 17: because all the points on the effective gain curve are below the optimal gain of the Snyder and Fishman estimator, it is deduced that we are applying too small a gain. This can be expressed as an error in modeling the driving noise as smaller than its actual value. The author chooses to disregard the alternative explanation of a larger effective R value

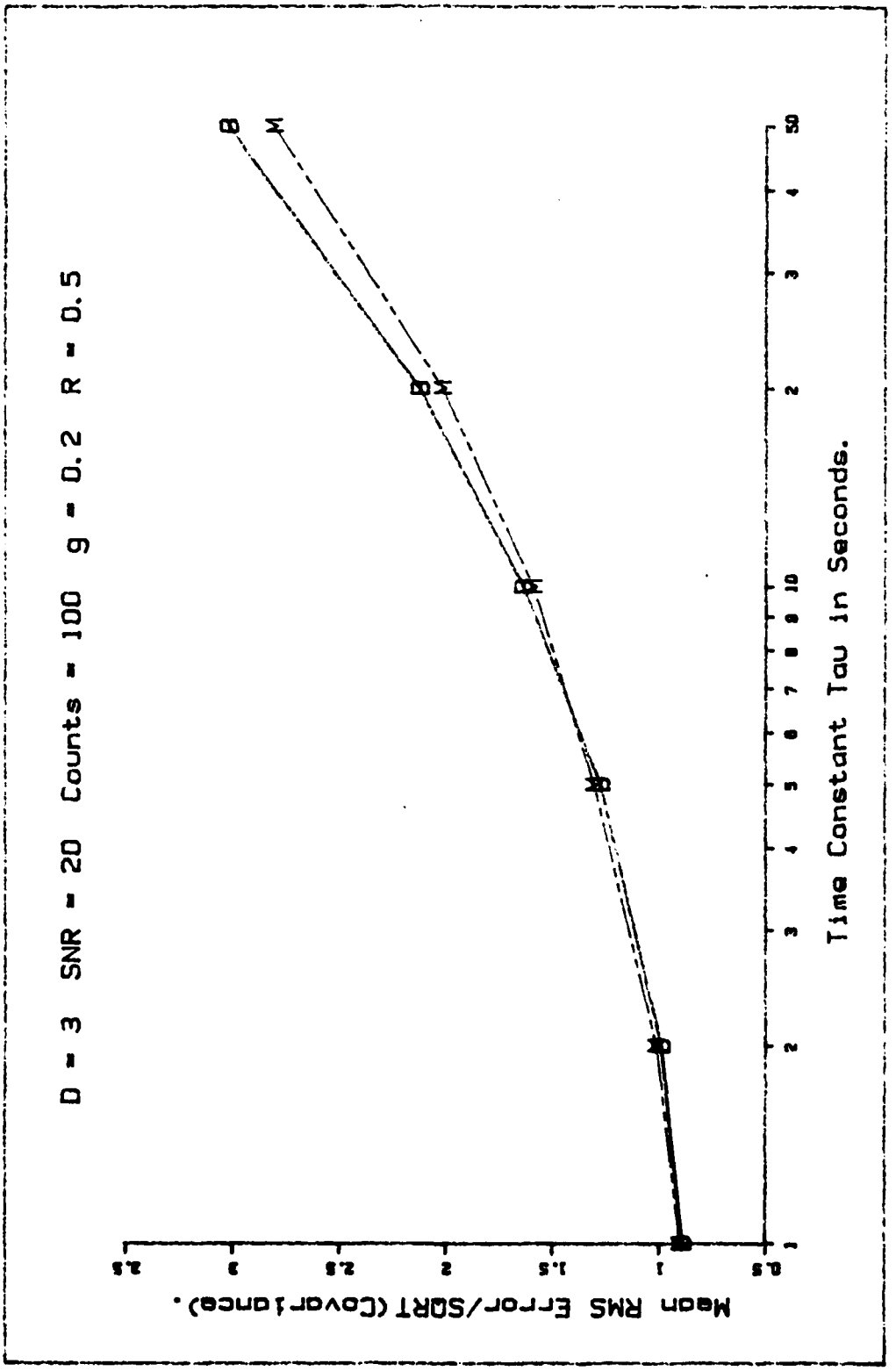


Figure 18. Estimator Accuracy Versus Tau.

because R also appears in the weighting factor Eqs (61) - (64), which tends to complicate the issue.

The explanation is begun by assuming that the probability distribution of the residuals about the beam centroid can be modelled as Gaussian with a standard deviation of $\sqrt{(\Sigma+R)}$; i.e. the residual covariance in the general case is given by $(\underline{H\Sigma H}^T+\underline{R})$. It is thus assumed that the density of the signal process residuals has the form

$$f_x(r-\hat{x}) = \exp\{-(r-\hat{x})^2/[2(\Sigma+R)]\}/\sqrt{2\pi(\Sigma+R)} \qquad (66)$$

Furthermore, by modeling the noise process residual distribution as uniform over the detector array we get

$$f_n(r-\hat{x}) = \begin{cases} 1/L & -L/2 < r < L/2 \\ 0 & \text{elsewhere} \end{cases} \qquad (67)$$

The total effective residual density function can therefore be defined by the weighted sum

$$\begin{aligned} f_t(r-\hat{x}) &= [f_s(r-\hat{x})]\left[\frac{\text{mean rate of arrival of signal}}{\text{mean rate of arrival of (signal+noise)}}\right] \\ &\quad +[f_n(r-\hat{x})]\left[\frac{\text{mean rate of arrival of noise}}{\text{mean rate of arrival of (signal+noise)}}\right] \\ &= [f_s(r-\hat{x})][SNR/(SNR+1)] \\ &\quad + [f_n(r-\hat{x})][1/(SNR+1)] \end{aligned} \qquad (68)$$

where the weighting is such that the total area under the new function is still unity. For the following calculations, the length of the detector array L is assume to be 10 cm and $\hat{x}$ is assumed to be zero. First the mean value of the weighting function as a function of the actual Σ is plotted as the upper solid line in Figure 19. This value is a ratio between the gain calculated in Eq (8) and an average or equivalent gain used in Eq (13) to update the estimator's calculation of the covariance:

$$K_m = W_m K = W_m \hat{\Sigma}(t_i^-)/(R+\hat{\Sigma}(t_i^-)) \tag{69}$$

where W_m is the mean value of the weighting factor defined in Eq (61), and

$$\hat{\Sigma}(t_i^+) = \hat{\Sigma}(t_i^-)(1-K_m) \tag{70}$$

Because of this lower equivalent gain, the estimated covariance of the estimator settles to a higher steady-state value than a Snyder and Fishman filter. This can be shown by solving Eq (11) for the scalar time-invariant case

$$\hat{\Sigma}(t_i^-) = \Phi^2 \Sigma(t_{i-1}^+) + \tau_g^2(1-\Phi^2)/2 \tag{71}$$

defining the steady state condition as $\hat{\Sigma}(t_{i-1}^+) = \hat{\Sigma}(t_i^+)$ and substituting in Eq (70) yields

$$\hat{\Sigma} = \Phi^2 \hat{\Sigma}(1-K_m) + \tau_g^2(1-\Phi^2)/2 \tag{72}$$

Combining this with Eq (69) yields the equation

$$0 = \Sigma(1-\Phi^2(1-W_m)) - \Sigma(R-\tau g^2/2)(1-\Phi^2) - \tau g^2 R(1-\Phi^2)/2 \qquad (73)$$

Since the mean weighting factor is relatively constant for Σ small with respect to R, a first approximation to the solution is to solve Eq (73) as a quadratic to yield the estimator covariance. The presence of the mean weighting factor can be seen to drive the steady state solution of the estimator covariance up from the Snyder and Fishman estimator solution where the weighting factor is unity. This value is used to calculate the gain applied by the estimator in updating the beam centroid position estimate. Because the gain applied to the residuals in Eq (12) is no longer independent of those residuals, the effective gain applied is not the same as that used in Eq (13). The effective gain ratio shown by the broken line in Figure 19 is the square root of the variance of the weighting function times the residual and divided by the filter-computed variance of the residuals as shown in the equation

$$W_v = \sqrt{E\{[(r-x)\{W(r-x)\}]^2\}}/(\Sigma+R) \qquad (74)$$

where the expectation is taken using the total residual density function defined in Eq (68). Note that the effective weighting factor W_v, applied to the gain used in updating the beam centroid position estimate according to Eq (11) is always smaller than that used to update the covariance calculation, so there is always some uncompensated uncertainty in the update calcu-

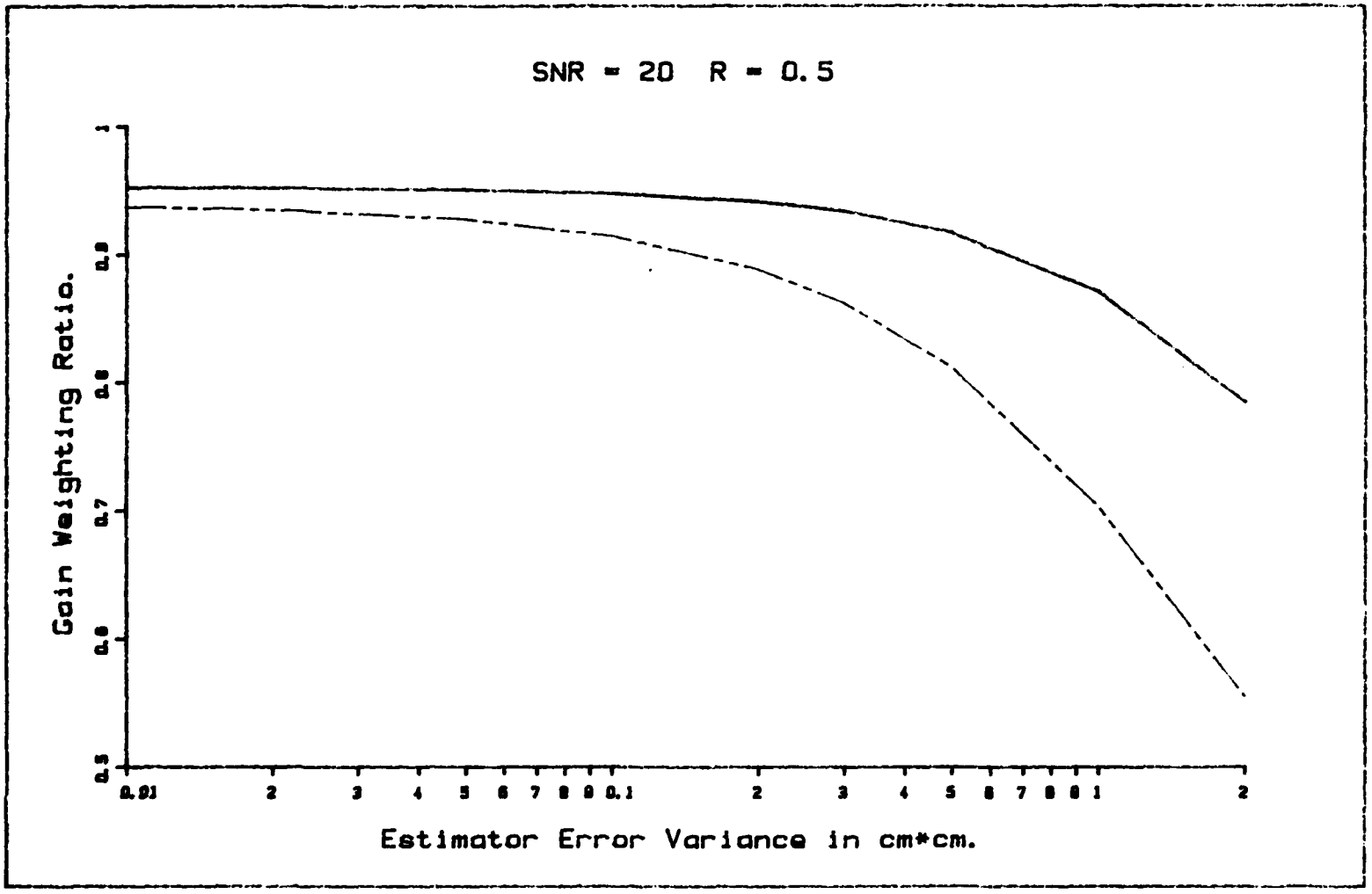


Figure 19. Weighted Gain Ratios.

lations. Using the true covariance between the filter and the centroid, Σ and W_v in Eqs (69) and (72) yields the actual error variance of the estimator as

$$\Sigma = \tau g^2(1-\phi^2)/\{2\cdot[1-\phi^2(1-W_vK)]\} \qquad (75)$$

As the time constant τ increases, this error tends to persist longer, causing the discrepency between the estimated performance and the actual performance of the estimator to increase, as was seen in Figure 18. However, this effect was not found to be large enough to account for the results seen, so the simulation software was examined more closely and the $\tau/2$ term was found to be missing from Eq (71) for the filter error variance calculations. But the term was present in the noise additions to the beam simulation, so that for the larger τ , there was a large mismatching of the driving noise modeled by the filter and that used in actuality. Because this error was discovered so late in the effort, it was not possible to rerun all the simulations with the corrected software (partial results are presented in the appendix), so it is necessary to keep in mind that as τ departs from 2 seconds, the performance of the estimator probably will be substantially better than that displayed in this thesis. However, though the effect of the weighting factors was discovered to be much smaller than it first appeared, it is still present and should be compensated

for. To counter this effect, it is possible to try adding a dynamics pseudo-noise to the system so that the gain weighting is compensated. The simulation was modified to allow this mismatching of the truth and estimator models, but the necessary runs to prove the effectiveness of the method were not accomplished by the time of this writing.

It is possible to derive similar effective weighting curves for the Best Half method by assuming that the weighted gain equals the Snyder and Fishman gain when the probability of the event being signal is greater than the probability that it originated from the noise process and is zero otherwise. These curves have higher values than the corresponding curves shown in Figure 19 but follow the same general trends except that the roll off observed as Σ becomes significant with respect to R begins at higher Σ values and is more sharply defined. It is interesting to note that the region in Figure 11 where the Merge method yielded better results than the Best Half method corresponds to this area in Figure 19, where the effective weighting curves begin to drop appreciably.

Regulator Performance

In this section, the effectiveness of the Meer filter in providing estimates of the beam centroid for use in regulating the beam position is evaluated. The regulator used for all these runs is calculated under the assumptions of a unity

weighting matrix U, and a control update period that is one-tenth of the beam centroid's time constant. In the figures associated with the following subsections, there will be three curves of the steady-state RMS value for the beam position under different control histories. The upper dashed line is associated with the uncontrolled beam (and is demarked by "U" for uncontrolled), and the lower broken line demarked with an "S" (for simulated state feedback) depicts the simulation results when the feedback is artifically drawn from state the beam simulation and corresponds to the theoretical best that may be hoped for from the regulator. The third control simulation, whose values are determined by the filters estimate of the beam state, is shown by the solid line (demarked with "F" for filtered state estimate feedback). The position of the solid line relative to the other two lines is indicative of the effectiveness of the filter in the control law. Because the Merge method yielded slightly better results in many cases, this was the method implemented in the estimator for the following simulations. Figure 20 shows the impact of the regulator on the time response of the beam centroid. Though the data point markings are omitted in the figure, the line types used are the same, so it can be seen that the controller with the estimator in the loop tends to overshoot its desired value. This effect arises from the transient behavior of the estimator and may probably be avoided by delaying the implementation of control feedback

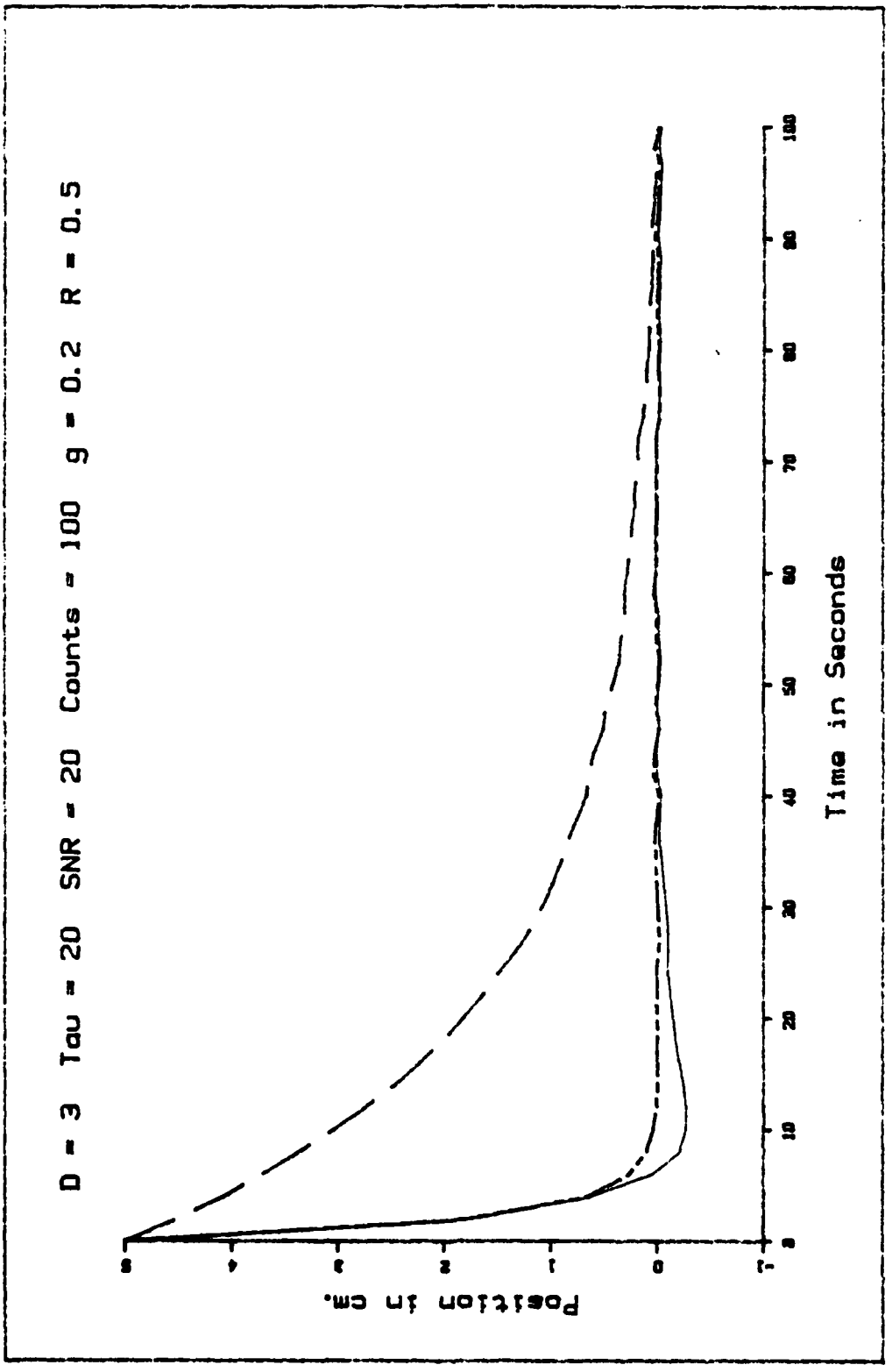


Figure 20. Effect of Regulator on Time Response.

until the estimation transients die out.

Regulator Performance Versus g. Figure 21 shows the relative performance of the control scheme as the dynamic driving noise is varied. Once again, the estimator model parameters are varied exactly as the true beam simulation parameters, so that the g used in both cases is identical. It should be noted that the asymptotic leveling off of the uncontrolled simulation curve represents the impact of average transient response that has not yet reached steady-state performance in only 2.5 time constants. To provide a truer indication of the steady state performance of the uncontrolled simulation, an additional dashed line is shown in Figure 21 which consists of an average over the last 10 seconds of the simulations. This was not done for the controlled curves because, as can be seen from Figure 20, the transients die out much more quickly for the regulated system. This tends to make the trend more apparent that the filter applied control is becoming less effective in relation to the effectiveness of full-state feedback control as the system becomes more deterministic with respect to the measurement parameter R. This trend seems physically reasonable: as g becomes smaller, it has less tendency to drive the beam centroid away from the center of the array so there is less error for the regulator to act upon, and the estimation errors tend to have a greater relative effect on the control applied. As g decreases the relative importance of the beam dispersion R

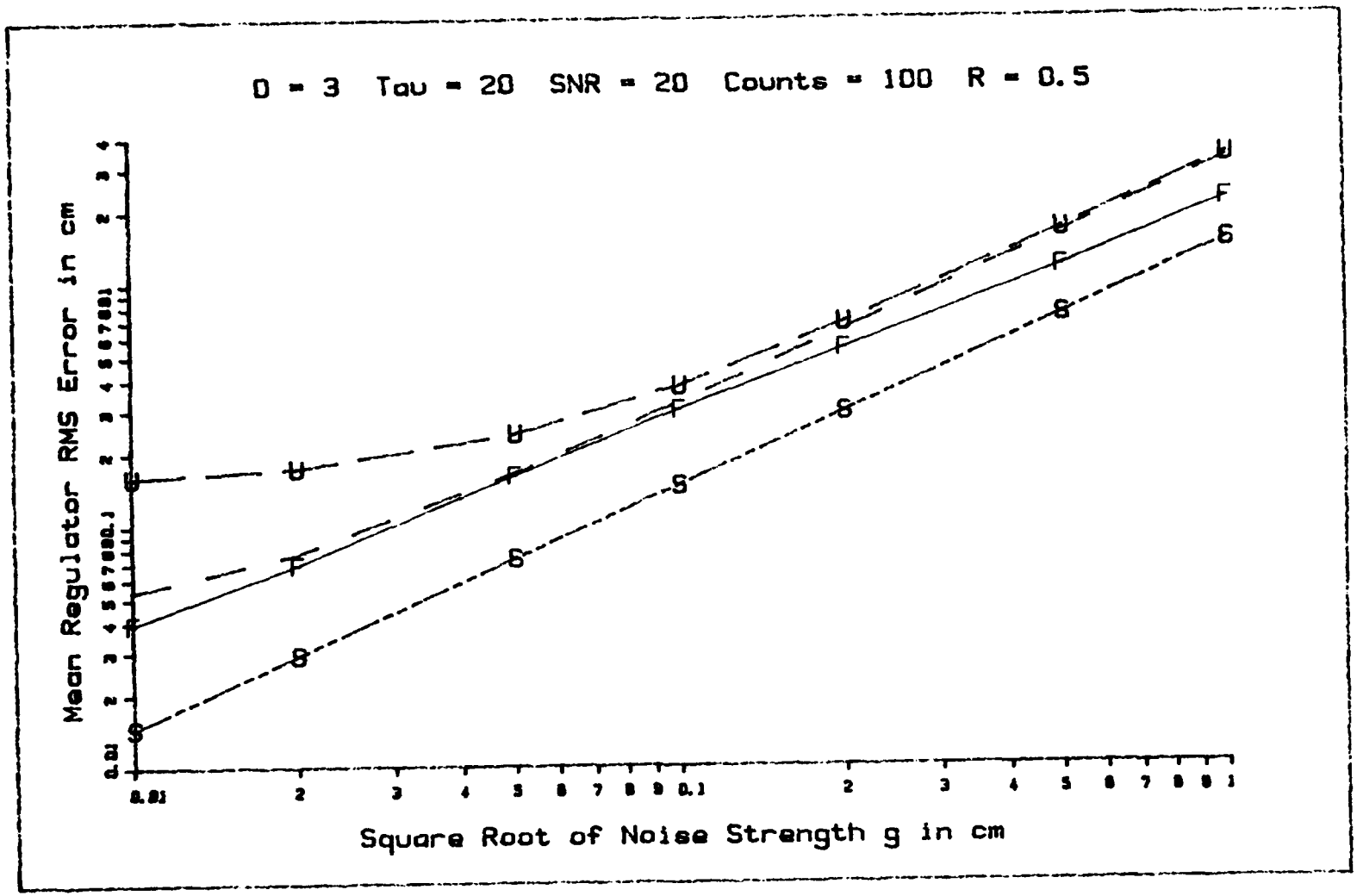


Figure 21. Regulator Performance Versus Dynamic Model Noise.

on the estimation errors increases, so that control based upon these relatively poorer estimates is as bad as no control at all.

Regulator Performance Versus R. Figure 22 displays the regulator performance as the beam dispersion R is varied. Note that the regulator's performance deteriorates as the beam dispersion increases. This is caused by the fact that as the beam dispersion increases, the weighting functions reject noise induced events less strongly. Thus, the estimator errors grow to the point where the control applied is ineffective in regulating the system.

Regulator Performance Versus Expected Number of Signal Events. The effect of the observed sampling rate on the regulator's performance is displayed in Figure 23. The ineffectiveness of the control scheme at the low sampling rate is as expected from looking at Figure 11, showing the estimator performance trends. Note that at 10 expected events, the controller actually does more harm than good! However, Figure 1. also suggests that a Best Half implementation might yield some improvement in the regulator performance at the lowest observed event rate. As the number of signal events increases, the estimator performance improves substantially, as does the effectiveness of the controller.

Regulator Performance Versus τ. Figure 24 displays the relative effectiveness of the control scheme as a function of τ. At least part of the divergence of the curves at the long-

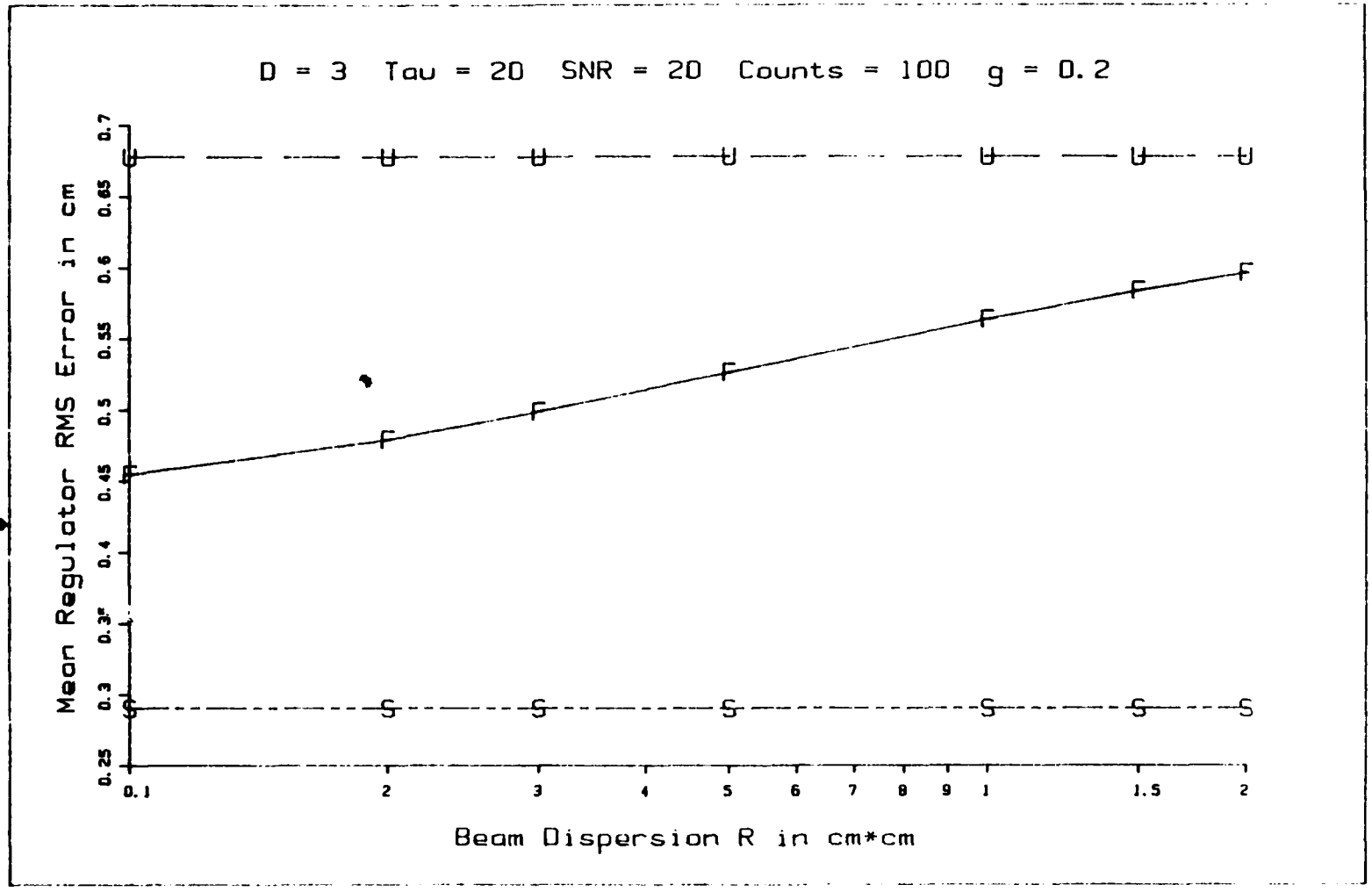


Figure 22. Regulator Performance Versus Beam Dispersion.

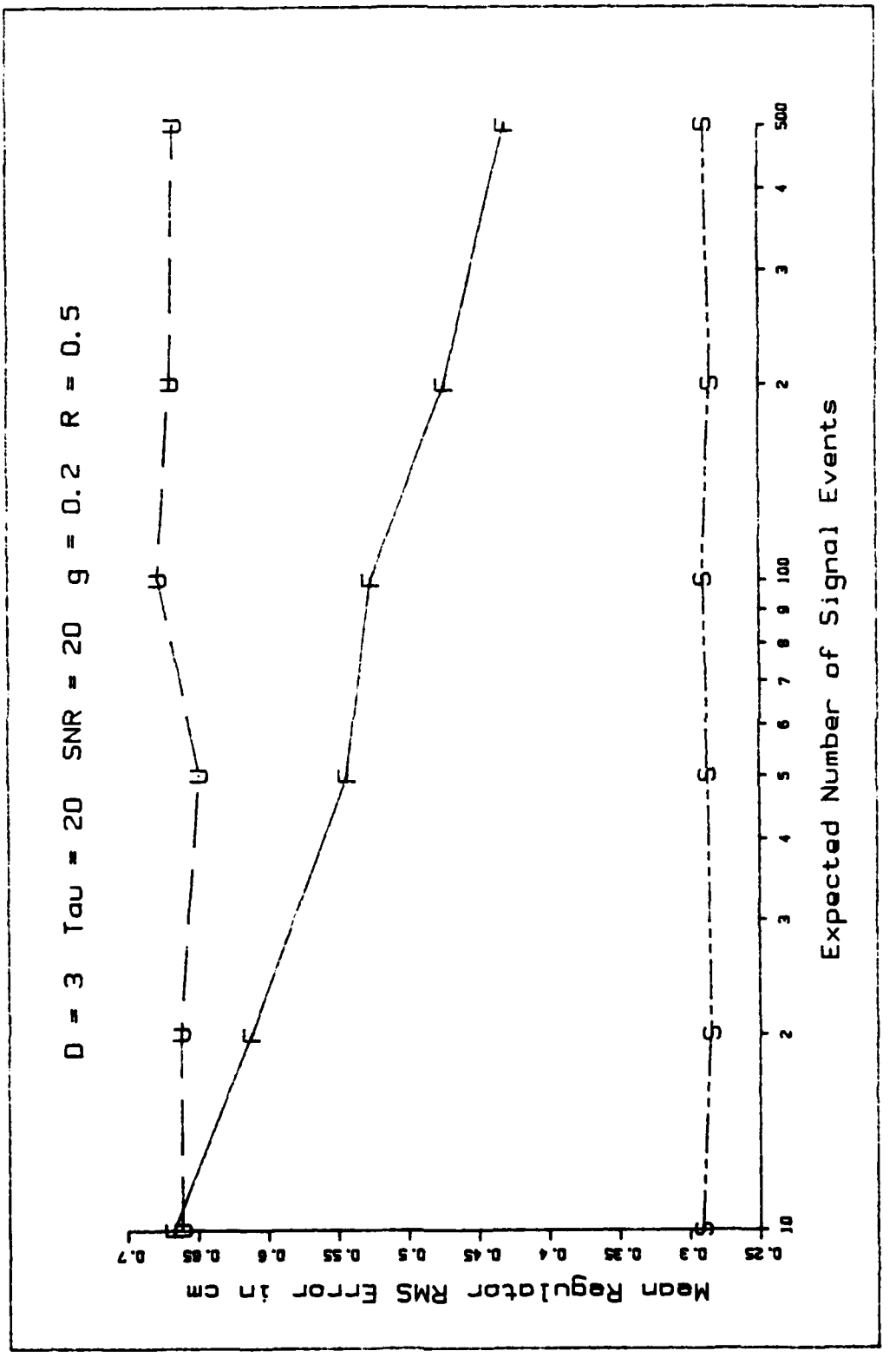


Figure 23. Regulator Performance Versus Expected Number of Signal Events.

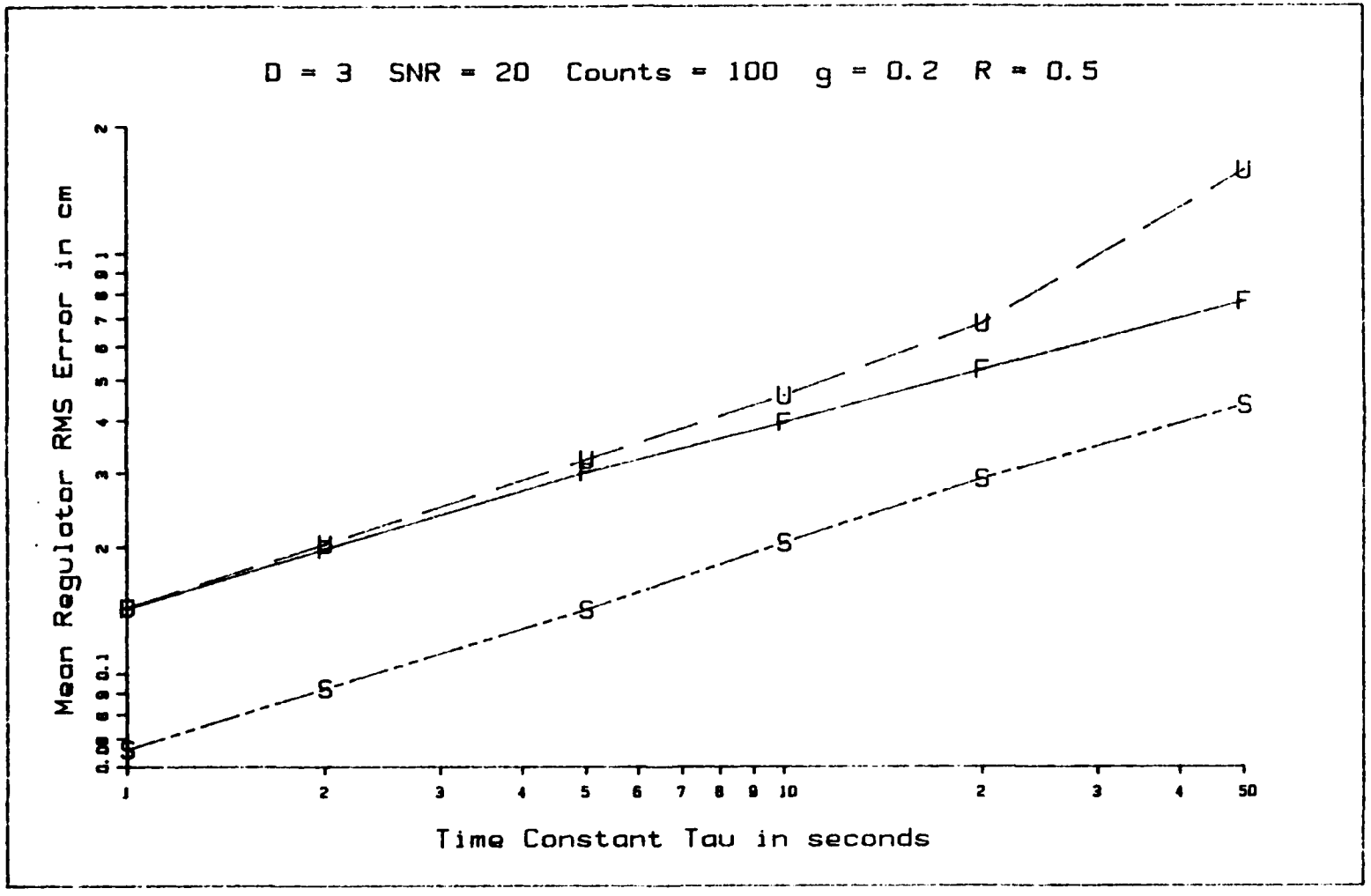


Figure 24. Regulator Performance Versus Tau.

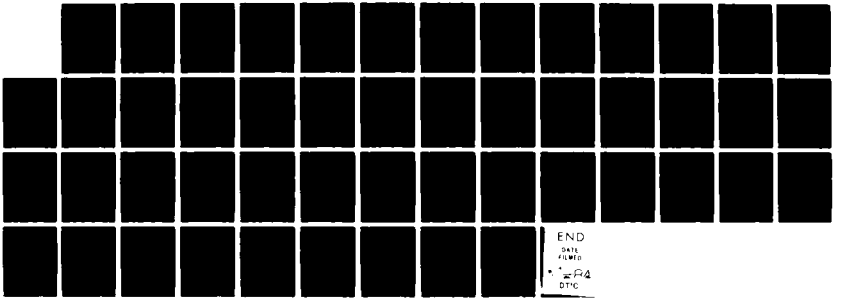
er time constants is once again due to the spurious contributions of the long settling time for the uncontrolled simulation curve. The ineffectiveness of the regulator for the smaller values of τ is driven by the fact that the average event rate used in these simulations is inadequate to provide good estimates of a system with the smaller values of the time constant, i.e., the system tends to look like a white noise source to the estimator. Thus as τ approaches zero, signal and noise events become less distinguishable, so that control based on the filter estimates becomes less effective. Because the filter tends to underestimate its own errors more severely for the larger τ cases (recall the discussion in the section on Tuning, as especially with regard to Figure 18), we can expect the effectiveness of the control to improve once some filter turning has been performed.

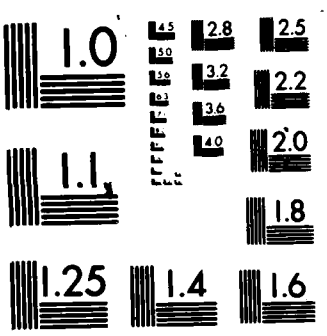
Regulator Performance Versus SNR. In Figure 25, it can be seen that varying the signal to noise count ratio has relatively little effect on the regulator performance. There is a slight improvement in the regulator's effectiveness as the signal to noise ratio increases, which is as expected from the filter performance trends. The time constant used in this run is large enough that we can expect an improvement in the estimator performance, and thus the regulator performance, by using the turning process discussed earlier.

Regulator Performance Versus D. It can be seen in Figure

AD-A138 311 POINTING AND TRACKING OF PARTICLE BEAMS(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING W L ZICKER DEC 83 AFIT/GE/EE/83-73 2/2

UNCLASSIFIED NL





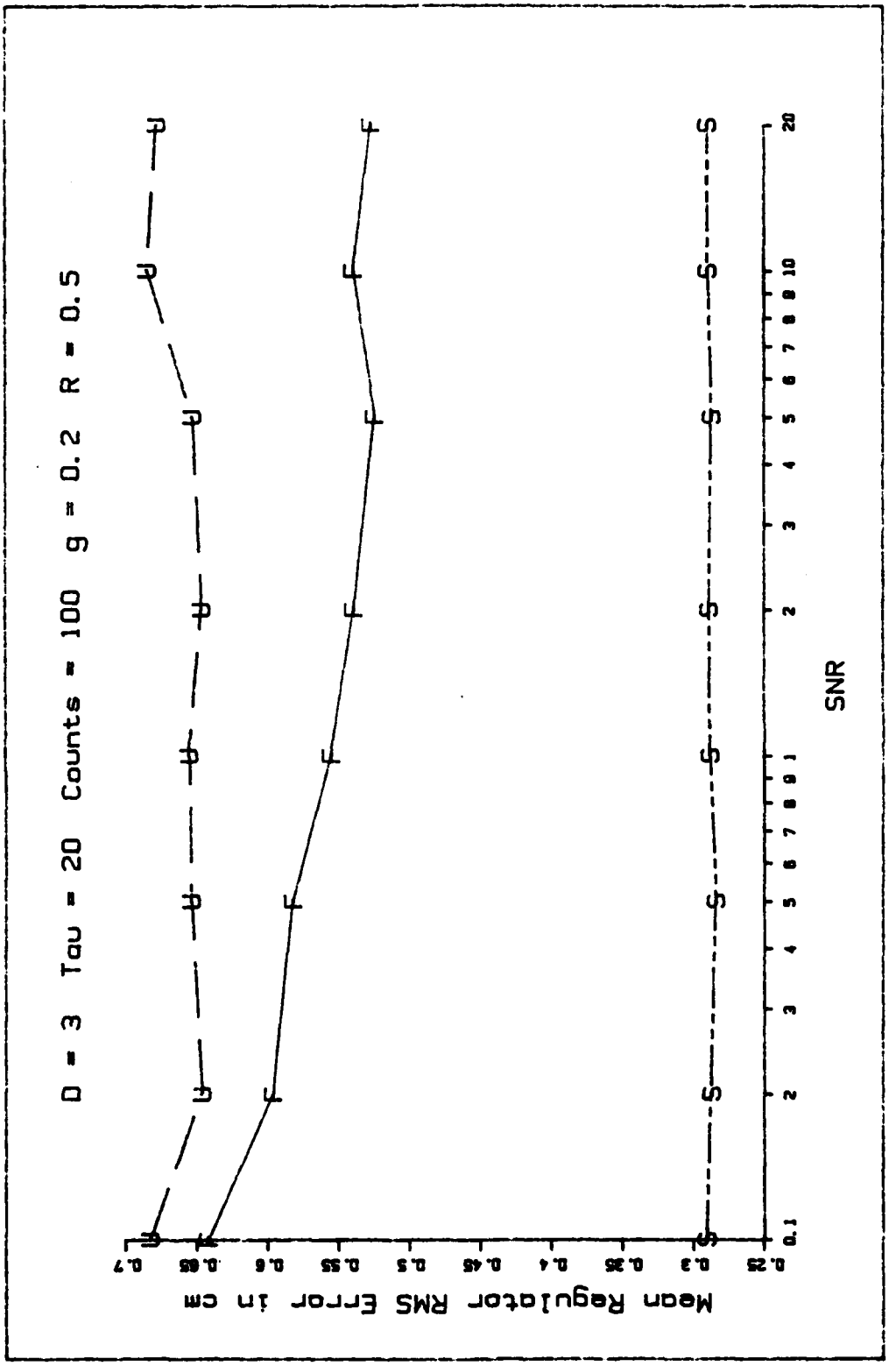


Figure 25. Regulator Performance versus Signal to Noise Count Ratio.

26 that the regulator's effectiveness is not a function of the depth of the estimator. Again, as in the filter section itself, this is closely tied to a scalar first-order Gauss-Markov model for the true beam centroid. It has been suggested that the depth of the estimator would be more important when higher dimension models are used. The relative ineffectiveness of the regulator is easily seen to be because the parameter set is in the region where the sampling rate is a poor match to the system time constant. More effective regulation performance would probably have been seen if another parameter set had been selected, but the author felt it was more consistent to continue to use the parameter set selected by Meer. However, it can be seen that the regulator is still having some effect even under the poor conditions chosen.

Tracker Performance

In this section, the effect of using the estimator in conjunction with a Kalman filter to track a target with the beam is examined. Because of the gain separation derived in Chapter III, only parameters associated with the target will be examined in this section. The beam parameters used in this section are the same as the design point used for most of the estimator and regulator evaluation done previously. The four parameters of concern separate into two groups; the first three parameters are perfectly modeled in the Kalman filter (i.e., as the real world parameters change, so do those in the filter),

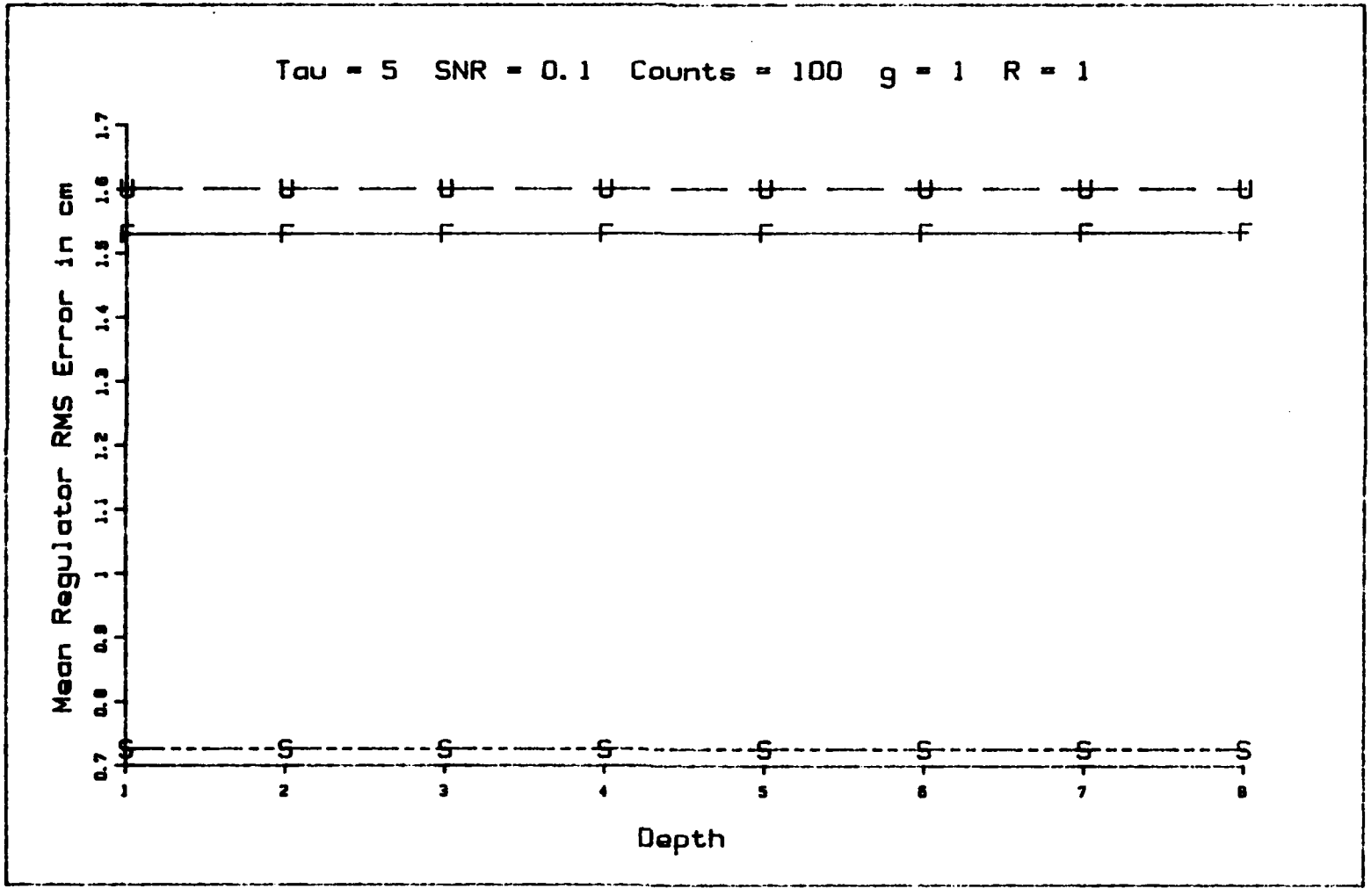


Figure 26. Regulator Performance Versus Depth.

the last parameter is the magnitude of a sinusoid driving the target state equation (42) whose frequency is calculated from equation (45) and which is not modeled by the filter. The target parameters about which the following sensitivity studies are accomplished are as follows: target time constant is 10 seconds so there is a different target transient characteristic than the beam centroid would follow, the target driving noise strength is 0.1 cm^2/sec, and G is unitless, the target measurement noise strength is set to 0.5 cm^2, and the magnitude of the unmodeled sinusoid is set to zero. The initial positions of the beam and the target are set at opposite ends of the detector array (plus and minus 5 cm) and are perfectly known by the filters. The initial uncertainty variances used in both estimators is 0.1 cm^2. Figure 27 shows the time response that may be expected with these parameters. The dashed line depicts the target simulation's time history, while the broken line which comes down and touches the target curve is the response of the beam when state values for control purposes are artificially drawn from the simulations. When the estimator and the Kalman filter are put into the closed loop, the response looks like the solid line which overshoots the target.

In the figures associated with the following sections, four lines are shown, each representing the performance results for different control schemes so that the relative impact of the estimator and the Kalman filter on the closed-loop system

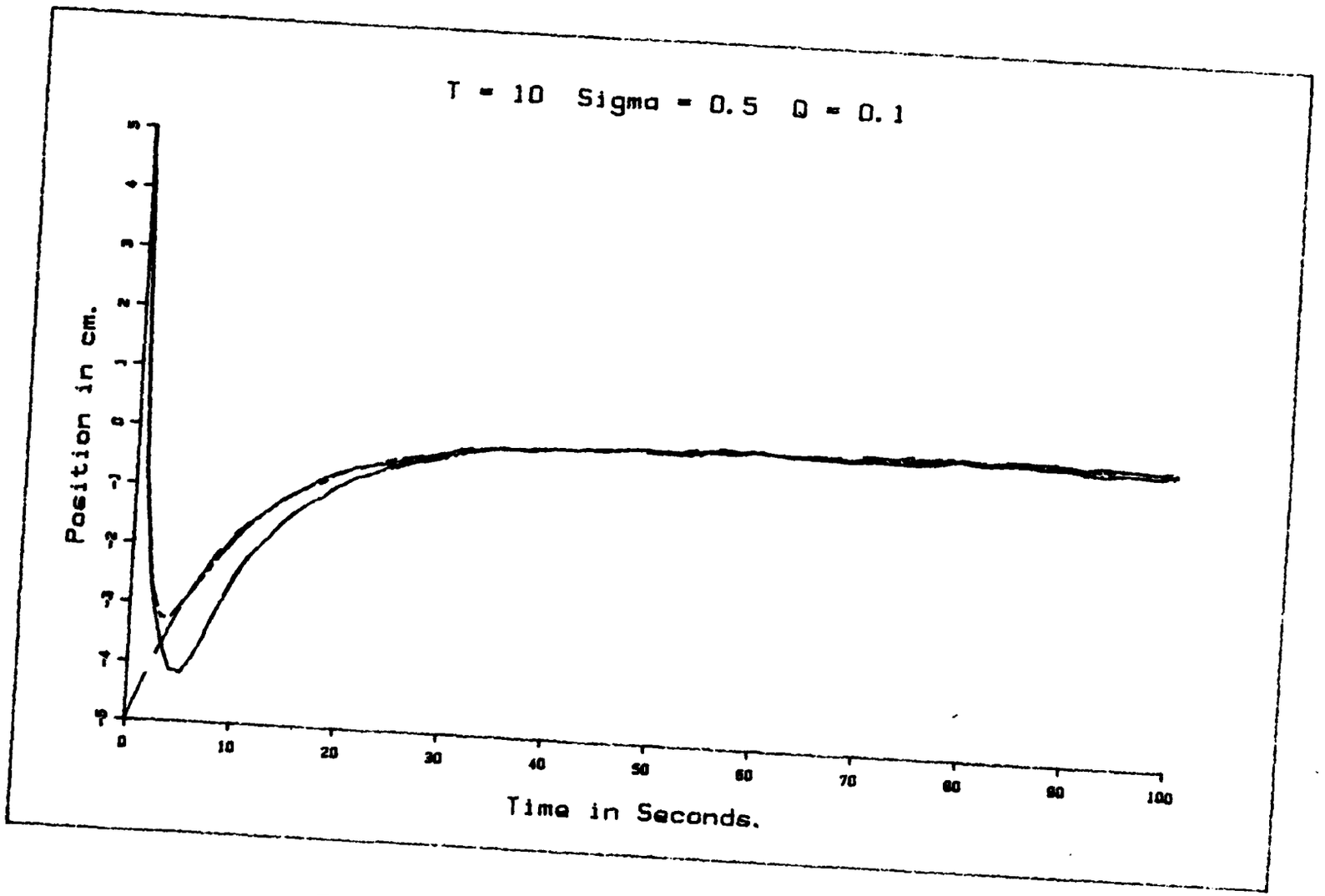


Figure 27. Tracker Time Response.

may be seen. The doubly broken line demarked by "S" represents the results obtained when both the beam and target states are artificially drawn from the simulation. The singly broken line demarked by "K" represents the use of the Kalman filter's estimate of the target state while still using the simulation's beam state for determining the control applied. The line broken by a dot and demarked by "M" shows the effect of using the Meer estimator's state and the simulation's target state. The solid line demarked by "F" depicts the performance results using both filters in the closed loop.

Sensitivity to Target Time Constant. Figure 28 shows the sensitivity of the closed-loop system to the target time constant. The values used for the target time constant T, were 2, 5, 10, 20, and 50 seconds. For these simulations, all parameters are identical in the filter and controller design to those used in the truth models. The control and Kalman filter update periods were primarily one tenth of the target time constant with exceptions at each end of the possible values. For the T=50 runs, the sampling periods were determined by the smaller beam time constant and set to 2 seconds rather than 5 for all four simulations. This explains the leveling off of the upward trend for the curves because the control period remains constant above T=20 seconds. At the T=2 value, a different problem arose: SOFE is set up to output data before and after Kalman filter updates, and because of system mass storage limits,

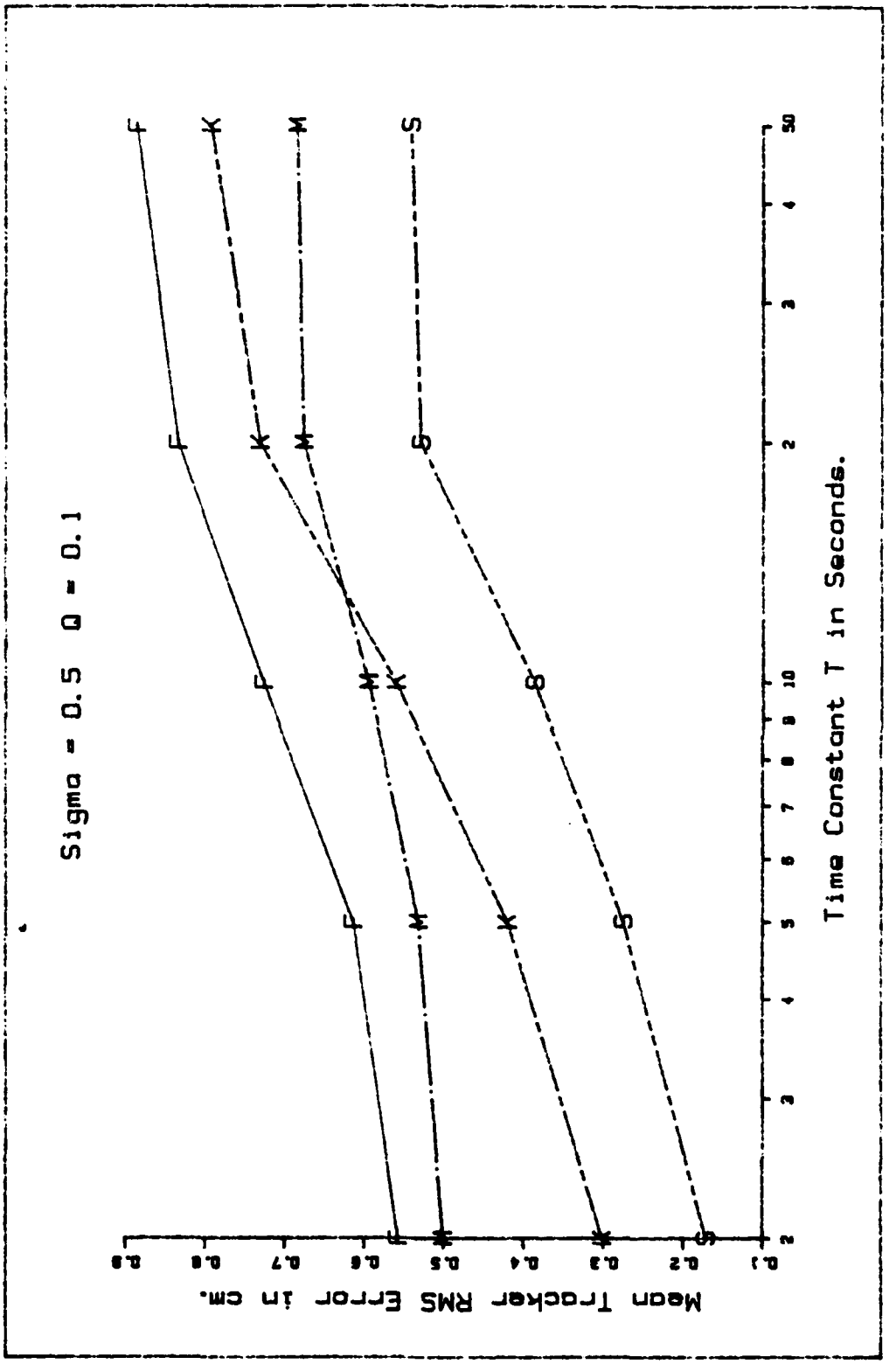


Figure 28. Tracker Performance Versus Target Time Constant.

it was not possible to update the Kalman filter every 0.2 seconds as desired. Therefore, the author settled for Kalman filter updates every 0.5 seconds while still adjusting the control value every 0.2 seconds because this update period was known to fit in the mass storage limits and because it was desired to attempt to maintain the control period's relationship with the target time constant. This mass storage limit represents a serious problem for future users of the software who may be interested in higher order system models unless some work is done to incorporate a simultaneous processor, rather than the postprocessor used by the author. The performance trends observed are as expected: as the target time constant increases, perturbations tend to persist longer so the Kalman filter becomes a more important factor in the overall performance of the system than the Meer estimator.

Sensitivity to Target Driving Noise. Figure 29 depicts the effect that varying the target driving noise strength has upon the ability of the controller to direct the beam at the target. As might be expected, increasing the strength of the target driving noise renders the target's position less predictable and thus makes the controller less effective. Of particular interest in this figure are the two partial estimator control curves. Note that for the lower end of the range of values where the target is relatively predictable, the Meer estimator errors are dominating the tracking errors. As the Kalman filter becomes less able to predict the target behavior,

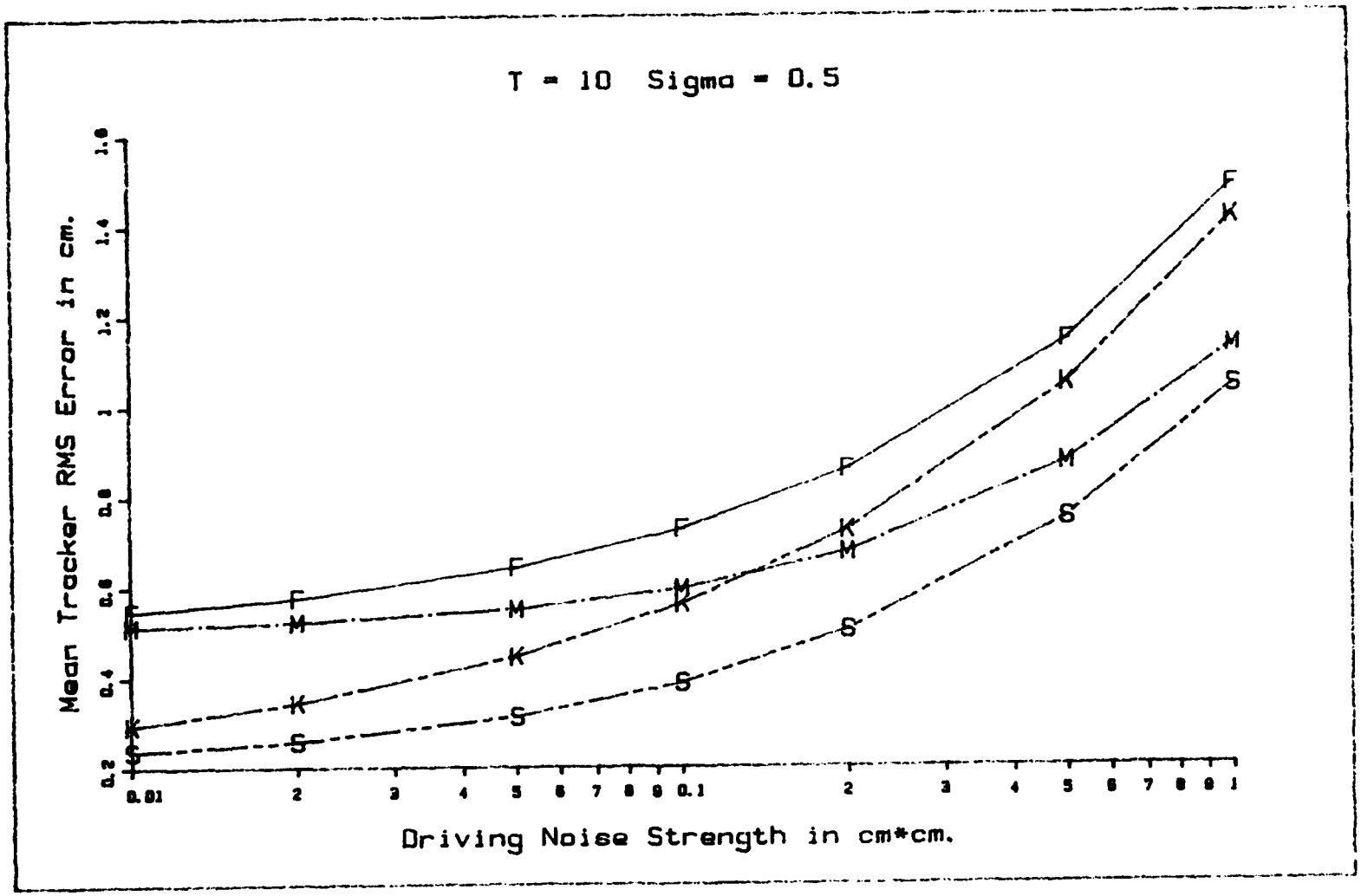


Figure 29. Tracker Performance Versus Target Driving Noise.

the Meer filter errors become less significant to the tracker's performance. This is analogous to the trend observed in Figure 28 of the previous section.

Sensitivity to Target Measurement Noise. In Figure 30, we see the effect that varying the strength of the measurement noise has on the tracking performance of the system. Because the measurement noise strength only affects the Kalman filter's estimates of the target state, the simulations in which the Kalman filter was not in the feedback loop are unaffected by its variation. Increasing the measurement uncertainty of course implies less certainty in the state estimates, causing the trend toward less effective control seen in the curves representing the incorporation of the Kalman filter's estimates into the feedback loop. This is analogous to the trend seen in Figures 28 and 29.

Sensitivity to Unmodeled Sinusoid. The simulation software was set up with an unmodeled sinusoidal driver in the target. The effect of this sinusoid on the system is shown in Figure 31. The dashed line again depicts the target time history, while the other lines represent the position of the beam centroid under the four different control modes. Note the fact that in steady-state the four curves merge into substantially two curves. The first of these, consisting of the response which follows the target most closely, consists of those simulations in which the Kalman filter does not appear in the control loop. The substantial difference in the steady-state gain

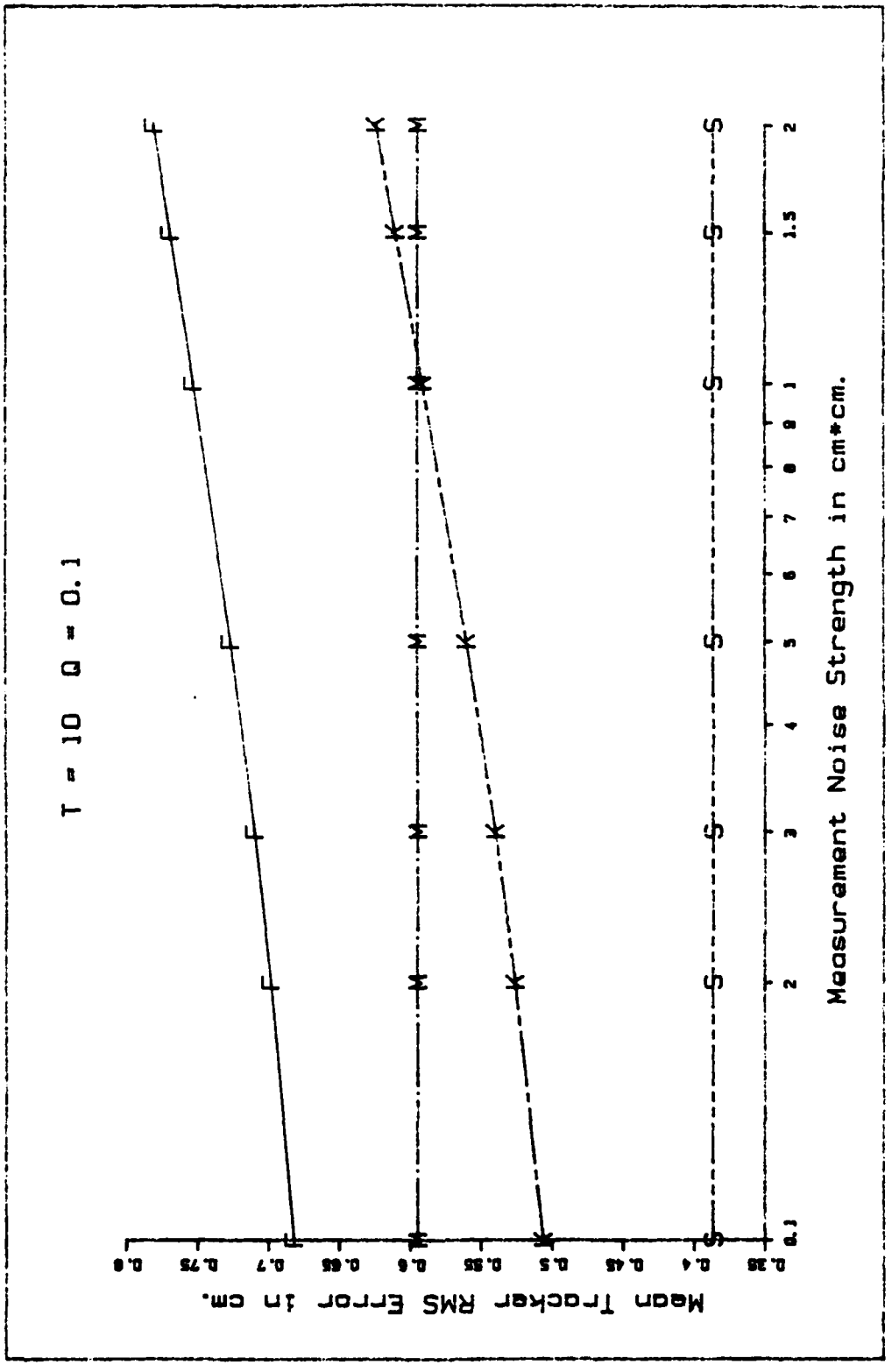


Figure 30. Tracker Performance Versus Target Measurement Noise.

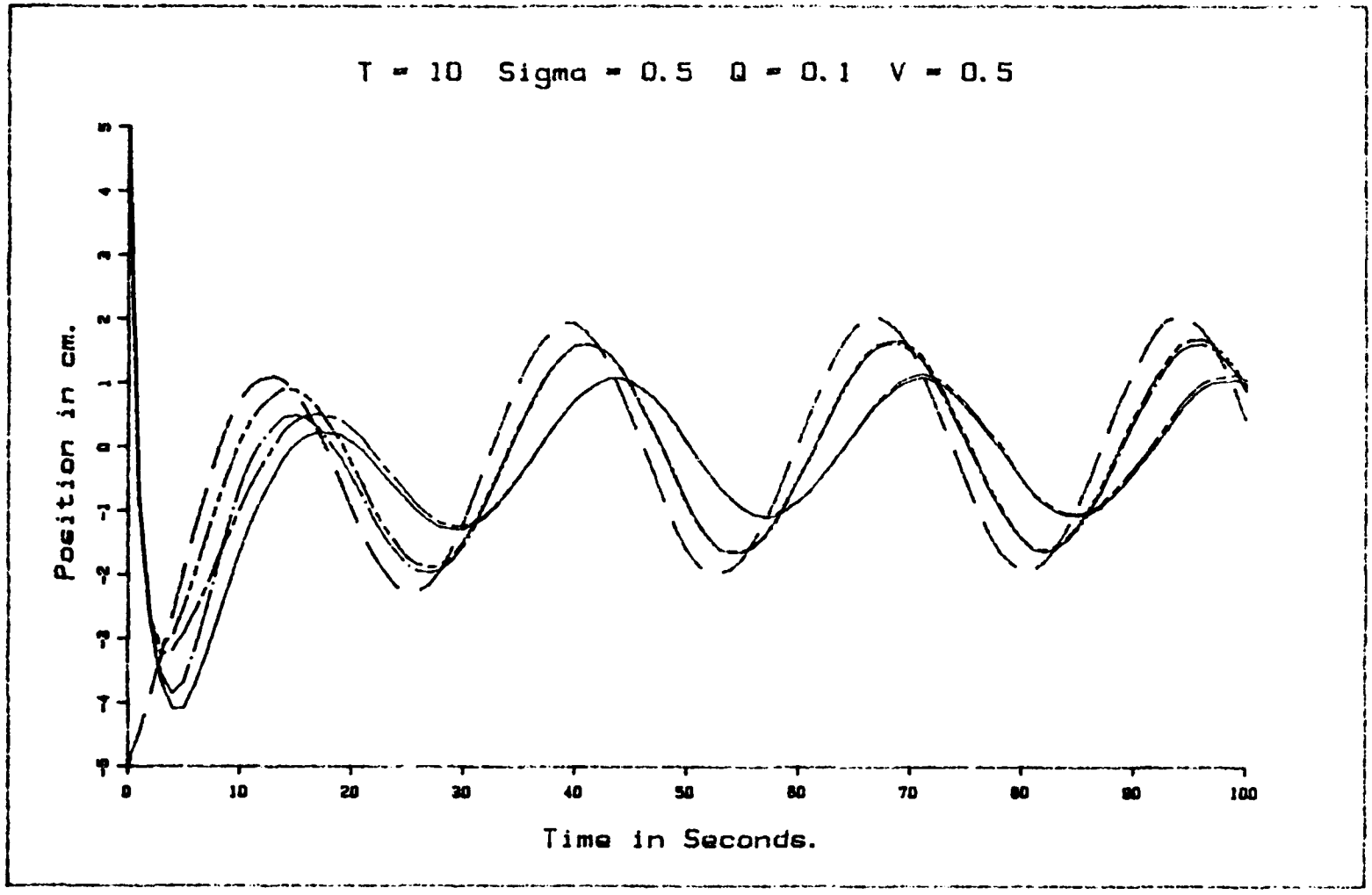


Figure 31. Tracker Time Response to Sinusoidal Input.

and phase shift suggest that substantially better tracking performance could be obtained with the same control feedback gain by substituting a Kalman filter based on a higher order target model. Perhaps using the sum of the current target state model and the output of an integrator driven by white noise, or a double integration driven by noise might be employed. However, these are astable shaping filters, and are therefore subject to introducing other potential problems, as in the solution to the Riccati equations for controller design (Ref 17 Vol III).

Figure 32 shows the sensitivity of the tracker to the unmodeled effect in a more quantitative manner. This figure shows that relatively small unmodeled effects do not have much impact upon the estimator performance, but when these effects become appreciable with respect to the Kalman filter's computed standard deviation, the system fails to track very well. This can be seen to be driving primarily by the inability of the Kalman filter to predict the target position. This reflects the fact that, once an estimator is in a control loop, robustness of the controller is degraded.

Summary

In this chapter, the performance of the Multiple Model Adaptive Estimator (MMAE) using a simple one-dimensional model for the beam centroid, under two possible implementations (i.e., Best Half and Merge), is examined and discussed. The performance sensitivities to the major parameters of a regula-

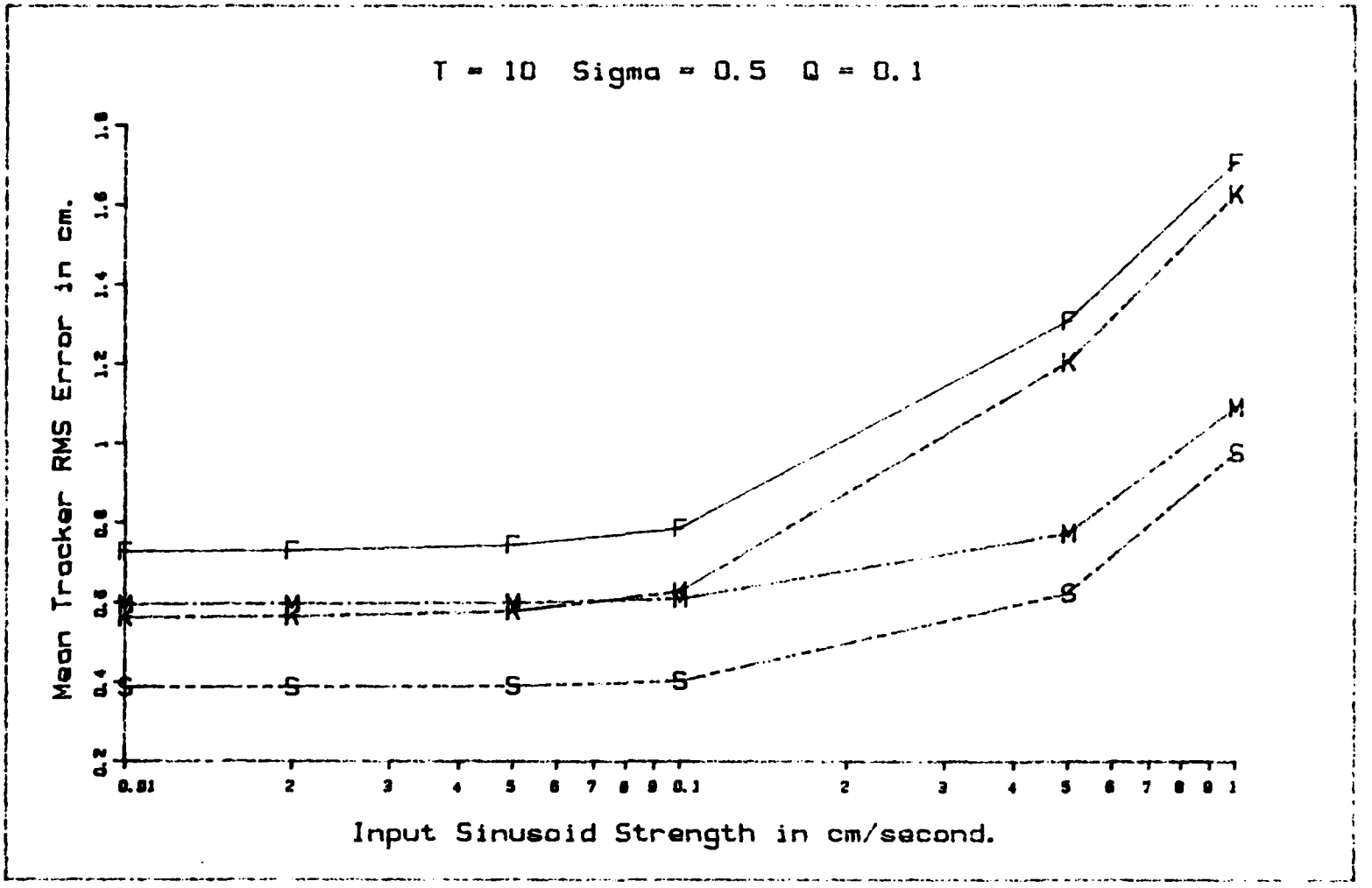


Figure 32. Tracker Performance Versus Unmodeled Sinusoidal Input.

tor and a tracker incorporating the MMAE are also presented and discussed.

The results are indicative of the overall performance that may be expected from using the MMAE in a control loop. This control methodology shows promise of providing significant control over the beam position over a wide range of parameter values. In the next chapter, the major conclusions derived from this research will be summarized and recommendations for further work will be presented.

VI Conclusions and Recommendations

Conclusions

The goal of this study was to determine the feasibility of using the Multiple Model Adaptive Estimator developed by Meer in a control loop. For this study, a simple LQG regulator and tracker were developed in Chapter III. The performance sensitivities of two possible implementations of the MMAE, Best Half and Merge, to the major parameters of the estimator were examined. This portion of the study revealed two major criteria to be of importance in considering the desirability of one method over the other. The first of these is that the Merge method provides better results than the Best Half method when the RMS performance of the estimator becomes large enough to be significant with respect to the beam width. The second factor to be considered is the relationship between the mean time between signal events and the time constant of the system. As this ratio approaches the Shannon sampling limit, the Merge method tends to deteriorate more rapidly than the Best Half method. However, at very high signal rates, the Merge method's additional computational requirements may prohibit its implementation despite the possibility of obtaining better performance, so there is presumably a range of signal rates for which the Merge method should be considered. The use of the weighting factors in the MMAE gives rise to two issues of interest to any-

one attempting to use the estimator. Because the weighting functions cause the estimator to reject events which have large residuals, the initial conditions of the estimator must be fairly close to the actual beam or the estimator will not acquire the beam (Alternatives are to increase the initial filter Σ; or if $\sqrt{R}$ is small with respect to the mismatching of $\hat{x}_0$ and x_0, other means must be found to assist acquisition). Also, because of this rejection of events with large residuals, the gain applied in updating the covariance calculations and that used to update the beam centroid are effectively different. This gives rise to the discovery that deliberately mismatching the driving noise parameter between a best description model of the dynamics and that actually used in the filter, can yield better performance for the estimator.

The feasibility of using the estimator in a control loop was demonstrated by the fact that in only one case did the regulated system fail to obtain better performance than the uncontrolled system. This was the case which presented the lowest information rate to the estimator, and there was a possibility that using the Best Half rather than the Merge method might still have resulted in some effective control of the system. At the design point, the regulator performance with the estimator in the loop is improved by approximately 40 percent of the range of improvement possible between the absence of control and the use of the regulator with perfect knowledge of the states.

Recommendations

The author feels that there is still a great deal of work to be done even with the relatively simple models used in the simulation for this research. The software should be modified to include the second method of calculating the weighting factors proposed, namely using the elemental rather than the overall residuals, so that the performance of this method may be compared to the method currently implemented. Two relatively simple modifications to the software, D=0 causing the use of a single Snyder and Fishman filter, and a negative signal to noise ratio turning off the noise process generator, would make the determination of performance bounds possible. Also, the simulation software currently includes a number of simplifications in the algorithm made possible by the scalar models used which should be generalized so that higher order models may eventually be examined. Moreover, the large number of Monte Carlo runs required to obtain good statistics caused mass storage problems in at least one of the tracker cases, so the software should also be modified to run the statistical accumulations simultaneously rather than in postprocessing mode, as currently accomplished via SOFEPL. Otherwise, the simulation will never be able to yield statistically meaningful results for higher order problems due to storage limitations. Also, the structure of the Snyder and Fishman filter is analogous to that of the Kalman filter and is thus subject to the

same numerical problems, so a factored form of the filter should be implemented (Ref 17 Vol I).

Additionally, how much improvement in the estimator performance can be gained by deliberately adding pseudo-noise to the elemental estimator covariance equation should be examined. This will yield information on the robustness of the estimator to mismatching this parameter; once the optimum performance point is established, the effect of mismatching the other parameters should also be examined to determine the robustness of the estimator/controller to realistic variations of the beam pointing and tracking environment.

The controllers implemented in this effort were a very simple LQG regulator and tracker and were only examined at a single weighting factor. The impact of varying the weighting factors on the system performance should be examined. Also, the effect of different controllers, such as proportional plus integral (PI) and command generator tracker (CGT) controllers (Ref 17, Vol III) should be available to the designer eventually using this simulation to decide upon a beam steering system design.

Bibliography

1. Athans, M. and C. B. Chang. "Adaptive Estimation and Parameter Identification Using Multiple Model Estimation Algorithm." Technical Note. M.I.T. 1976-28, 23 Jun 1976. AD-A028510.

2. Athans, M., D. Castanon, K. Dunn, C. Greene, W. Lee, N. Sandell, and A. Willsky. "The Stochastic Control of the F8-C Aircraft Using the Multiple Model Adaptive Control (MMAC) Method-I: Equilibrium Flight." IEEE Trans. AC: 768-780 (Oct 77)

3. Athans, M., R. H. Whiting, and M. Gruber. "A Suboptimal Estimation Algorithm with Probabilistic Editing for False Measurements with Applications to Target Tracking with Wake Phenomena." IEEE Trans. AC 22-3: 372-384 (Jun 77).

4. Athans, M. and D. Willner. "A Practical Scheme for Adaptive Aircraft Flight Control Systems." NASA-TN-D-7647, p. 315-336. MIT, 1973. AD 760790.

5. Chen, Chi-Tsong, Analysis and Synthesis of Linear Control Systems. New York: Holt, Rinehart and Winston, Inc., 1975.

6. Deshpande, J. G., T. N. Upadhyay, and D. G. Lainiotis. "Adaptive Control of Linear Stochastic Systems," Automatica, 9: 107-115 (1973).

7. Dorato, P. and A. H. Levis. "Optimal Linear Regulators: The Discrete Time Case." IEEE Trans. AC 16-6: 613-620 (Dec 71).

8. Eisberg, R. M. Fundamentals of Modern Physics. New York: John Wiley and Sons, Inc., 1961.

9. Fegley, K. A., S. Blum, J. O. Bergholm, A. J. Calise, J. E. Marowitz, G. Porcelli, and L. P. Sinha. "Stochastic and Deterministic Design and Control via Linear and Quadratic Programming." IEEE Trans. AC 16-6: 759-766 (Dec 71).

10. Fishman, P. M. and D. L. Snyder. "The Statistical Analysis of Space-Time Point Processes." IEEE Trans. IT 22-3: 257-274 (May 76).

11. Lainiotis, D. G. "Optimal Adaptive Estimation: Structure and Parameter Adaptation." IEEE Trans. AC-16: 160-170 (Apr 71).

12. -----. "Joint Detection, Estimation and System Identification." Information and Control 19: 75-92 (1971).

13. -----. "Sequential Structure and Parameter-Adaptive Pattern Recognition - Part I: Supervised Learning." IEEE Trans. IT, 16-5: 548-556 (Sep 70).

14. -----. "Supervised Learning Sequential Structure and Parameter Adaptive Pattern Recognition: Discrete Data Case." IEEE Trans. IT: 106-110 (Jan 71).

15. Levine, W. S., T. L. Johnson, and M. Athans. "Optimal Limited State Feedback Controllers for Linear Systems." IEEE Trans. AC 16-6: 785-792 (Dec 71).

16. Magill, D. T. "Optimal Adaptive Estimation of Sampled Stochastic Processes." IEEE Trans. AC 10-4: 434-439 (Oct 65).

17. Maybeck, P. Stochastic Models, Estimation, and Control. New York: Academic Press, 1979 (Vol. I), 1982 (Vols. II and III).

18. -----. "Design and Performance Analysis of an Adaptive Extended Kalman Filter for Target Image Tracking." AGARD-AG-256: (Mar 82).

19. Maybeck, P. and D. E. Mercier. "A Target Tracker Using Spatially Distributed Infrared Measurements." IEEE Trans. AC 25-2: 222-225 (Apr 80).

20. Maybeck, P. and S. K. Rogers. "Adaptive Tracking of Dynamic Multiple Hot-Spot Target IR Images," IEEE Trans. AC 28-10: 937-943 (Oct 83)

21. Maybeck, Peter S., Worsley, William H., and Flynn, Patrick M., "Investigation of Constant Turn-Rate Dynamics Models in Filters for Airborne Vehicle Tracking," NAECON Vol. 2: (May 82).

22. Meer, D. E., "Multiple Model Adaptive Estimation for Space-Time Point Process Observations," PhD dissertation, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, September 1982.

23. Murphy, D. J. "Batch Estimation of a Jump in the State of a Stochastic Linear System." IEEE Tran. AC: 275-276 (Apr 77).

24. Musick, S. H. SOFE: A Generalized Digital Simulation for Optimal Filter Evaluation User's Manual, AFWAL Technical Report 80-1108, Wright-Patterson AFB, Ohio: Air Force Wright Aeronautical Laboratories, 1980.

25. Musick, S. H., R. E. Feldmann, and J. G. Jensen, SOFEPL: A Plotting Postprocessor for 'SOFE', User's Manual, AFWAL Technical Report 80-1108, Wright-Patterson AFB. Ohio: Air Force Wright Aeronautical Laboratories, 1981.

26. Papoulis, A. Probability, Random Variables, and Stochastic Processes. New York: McGraw-Hill, 1965.

27. Price, W. J. Nuclear Radiation Detection. New York: McGraw-Hill, Inc., 2ed. 1964

28. Ogata, Katsuhiko. Modern Control Engineering. New Jersey: Prentice-Hall, Inc., 1970.

29. Santiago, J. "Fundamental Limitations of Optical Trackers," Master's Thesis, AFIT, Wright-Patterson AFB, Ohio, 1978.

30. Sanyal, P., and C. N. Shen. "Bayes Decision Rule for Rapid Detection and Adaptive Estimation with Space Applications," IEEE Trans. AC: 228-231 (Jun 74).

31. Sims, F. L. and D. G. Lainiotis. "Recursive Algorithm for the Calculation of the Adaptive Kalman Filter Weighting Coefficients." IEEE Trans. AC: 215-218 (Apr 69).

32. Smith, P. and G. Buechler. "A Branching Algorithm for Discriminating and Tracking Multiple Objects," IEEE Trans. AC: 101-104 (Feb 75).

33. Snyder, D. L., Random Point Processes. New York: John Wiley and Sons, Inc., 1975.

34. Snyder, D. L., and P. M. Fishman. "How to Track a Swarm of Fireflies by Observing their Flashes," IEEE Trans. IT: 692-695 (Nov 1975)

35. Tse, E. "On the Optimal Control of Stochastic Linear Systems," IEEE Trans. AC 16-6: 776-784 (Dec 71).

36. Weiss, J. L., T. N. Upadhyay, and R. Tenney. "Finite Computable Filters for Linear Systems Subject to Time Varying Model Uncertainty," paper presented at NAECON May 83 of results obtained in a Masters Thesis by Jerold L. Weiss, "A Comparison of Finite Filtering Methods for Status Directed Processes," Massachusetts Institute of Technology, Cambridge, Massachusetts, June 1983

37. Wenk, C. J. and Y. Bar-Shalom. "A Multiple Model Adaptive Control Algorithm for Stochastic Systems with Unknown Parameters." IEEE Conference on Decision and Control Proc.: 723-730, Fort Lauderdale, FL, (1979).

38. Willsky, A. S. "A Survey of Design Methods for Failure Detection in Dynamic Systems." Automatica, 12: 601-611 (1976).

Appendix

Parital Results with Corrected Software

At the end of this effort, an error was discovered in the simulation software which caused a mismodeling between the noise strength used to simulate the beam and that used by the estimator. Because of the lateness of this discovery it was not possible to repeat the full analysis with the corrected software. However, it was possible to run a more limited set of simulations to determine the relative effect of the correction.

The following set of figures are numbered to correspond with those used in the main text. They are also drawn as much as possible to the same scale so that it is possible to overlay them for direct comparison.

At the design point, there is a 25% improvement in the estimator's performance and most of the observed filter mistuning is accounted for by the correction. The difference between the regulator performance obtained with the estimator in the loop and that obtained using artifically exact knowledge of the beam centroid decreases by about one-third.

Also included are some additional figures starting at Figure 33' which show the trends of the mismatching of the RMS error and the square root of the filter-computed vari-

ance. The basic trends observed in the main text are still present in these results but with different numberical values.

Of particular interest is the trend shown in Figure 36' depicting the RMS error of the best half estimator divided by the square root of its filter-computed variance, which was previously masked by the effects of the software error. Note that the curve rises to a peak and then falls off, reflecting a mismatching of the estimator to the problem that can be corrected by tuning the estimator. The effect of this tuning on the curve in Figure 13' would probably be to make the slope steeper below this peak and milder above the peak. It is felt that taking more data would allow placing the peak near a SNR value of one-third which corresponds to the inverse of the estimator depth of three for this simulation. This should be confirmed by repeating this analysis for different depths. If it is confirmed, the result is a rule of thumb design criterion stating that the depth of the estimator should be greater than the inverse of the signal-to-noise count ratio expected. A corrollary of this criterion is that for SNR values greater than unity all that is required is a simple weighted gain version of the elemental Snyder and Fishman estimator.

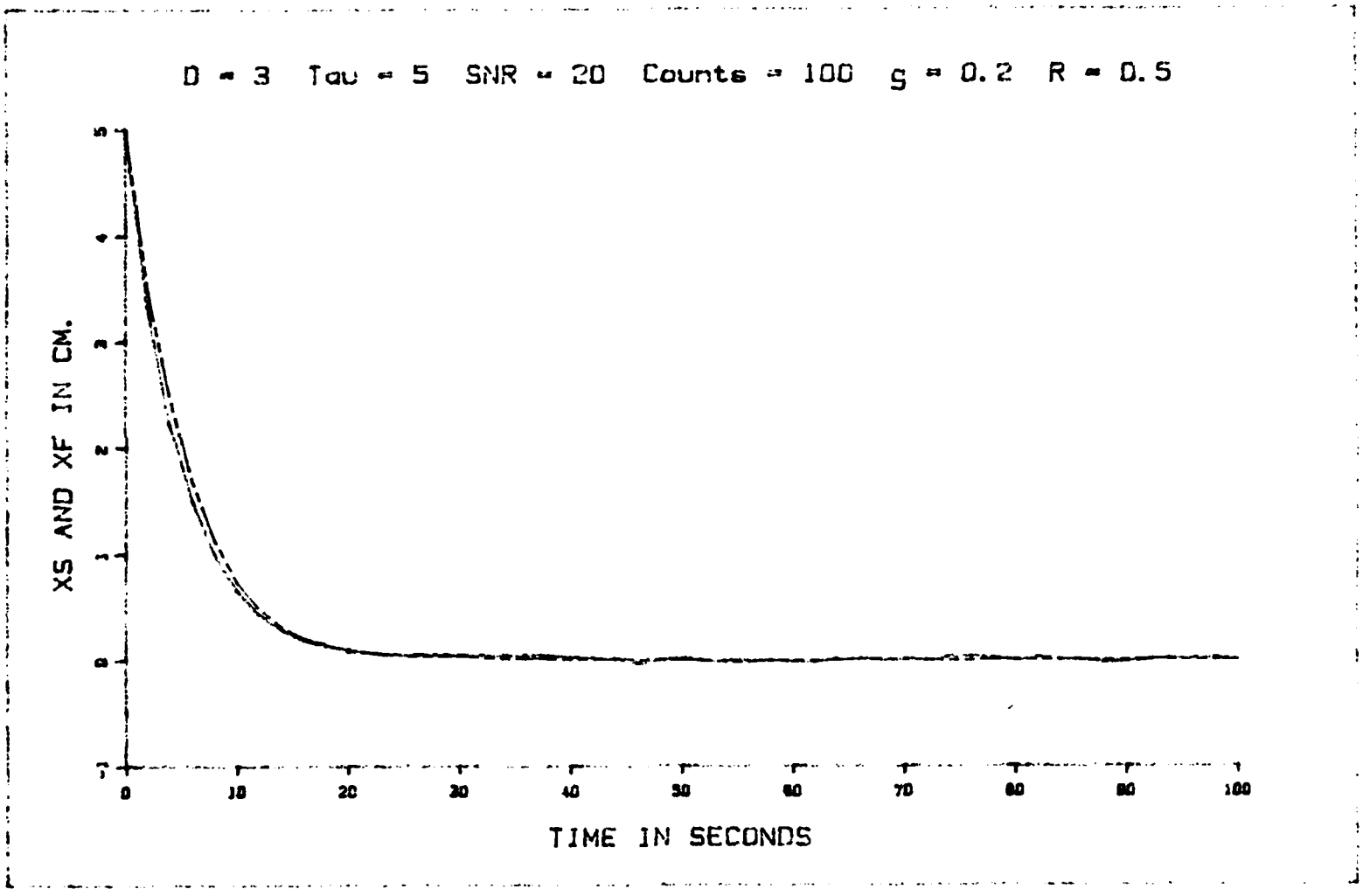


Figure 7'. True and Estimated Ensemble Averages for 200 Runs.

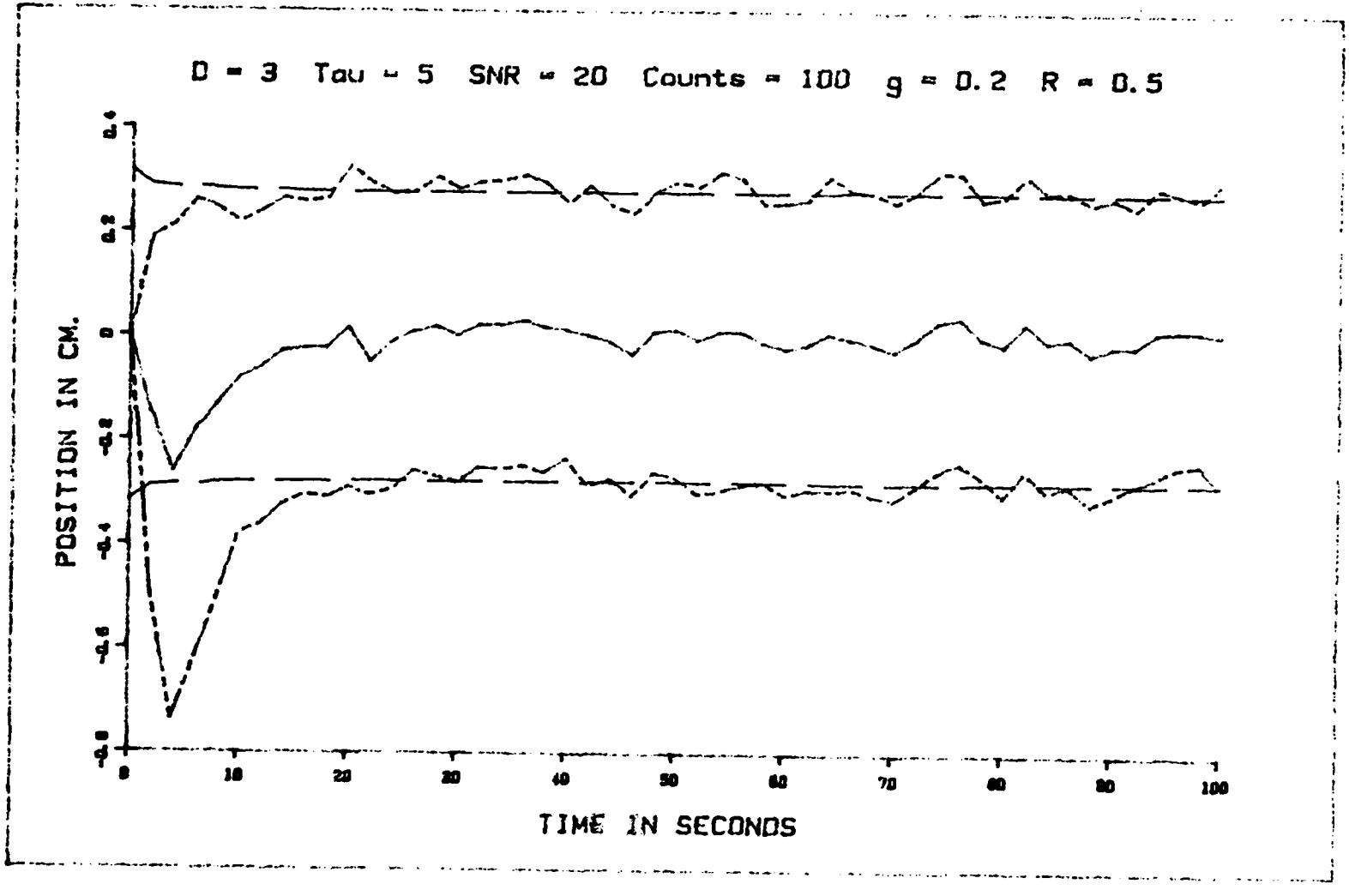


Figure 8'. Ensemble Error Statistics for 200 Runs.

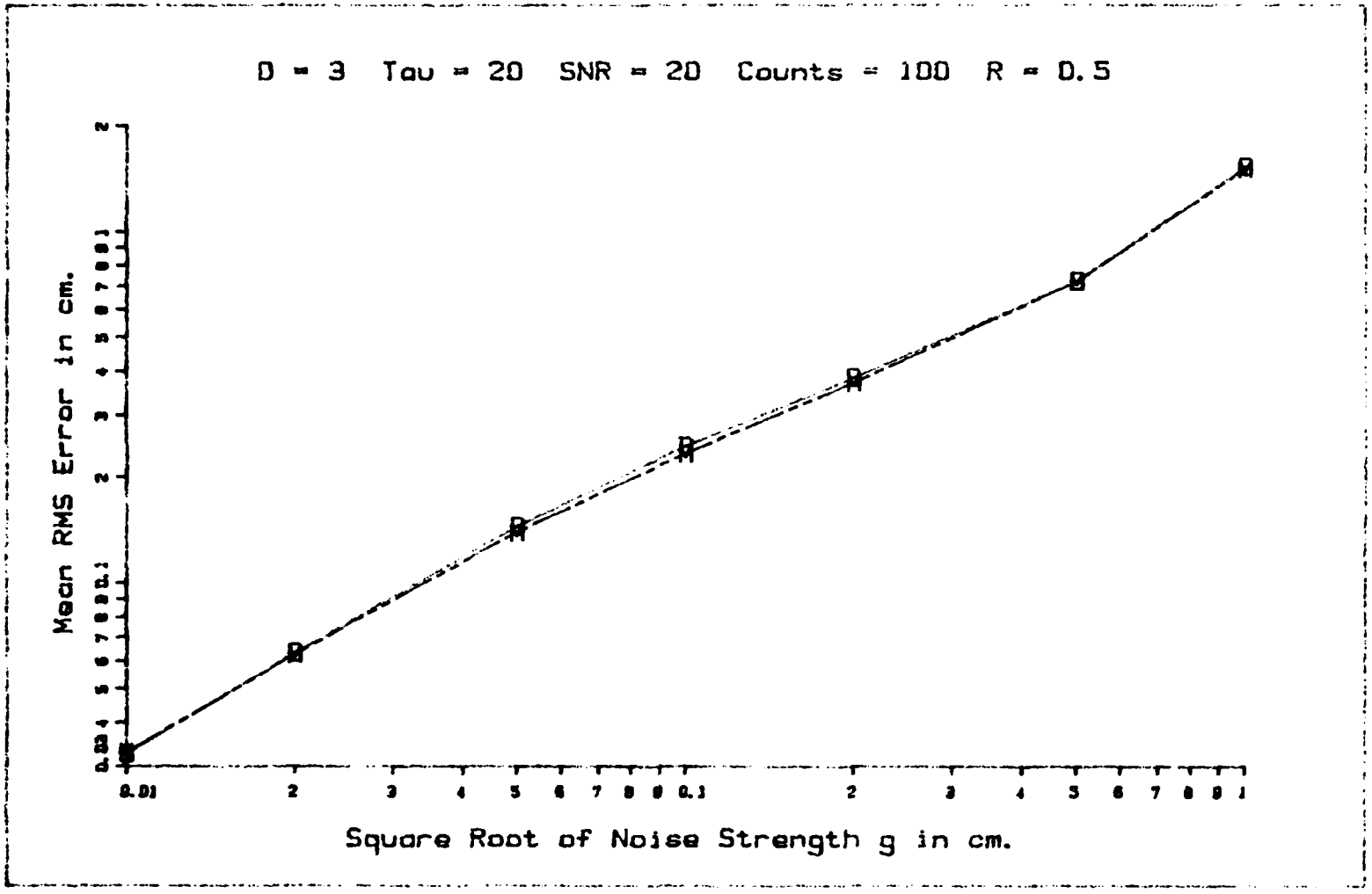


Figure 10'. Estimator Performance Versus Dynamic Model Noise.

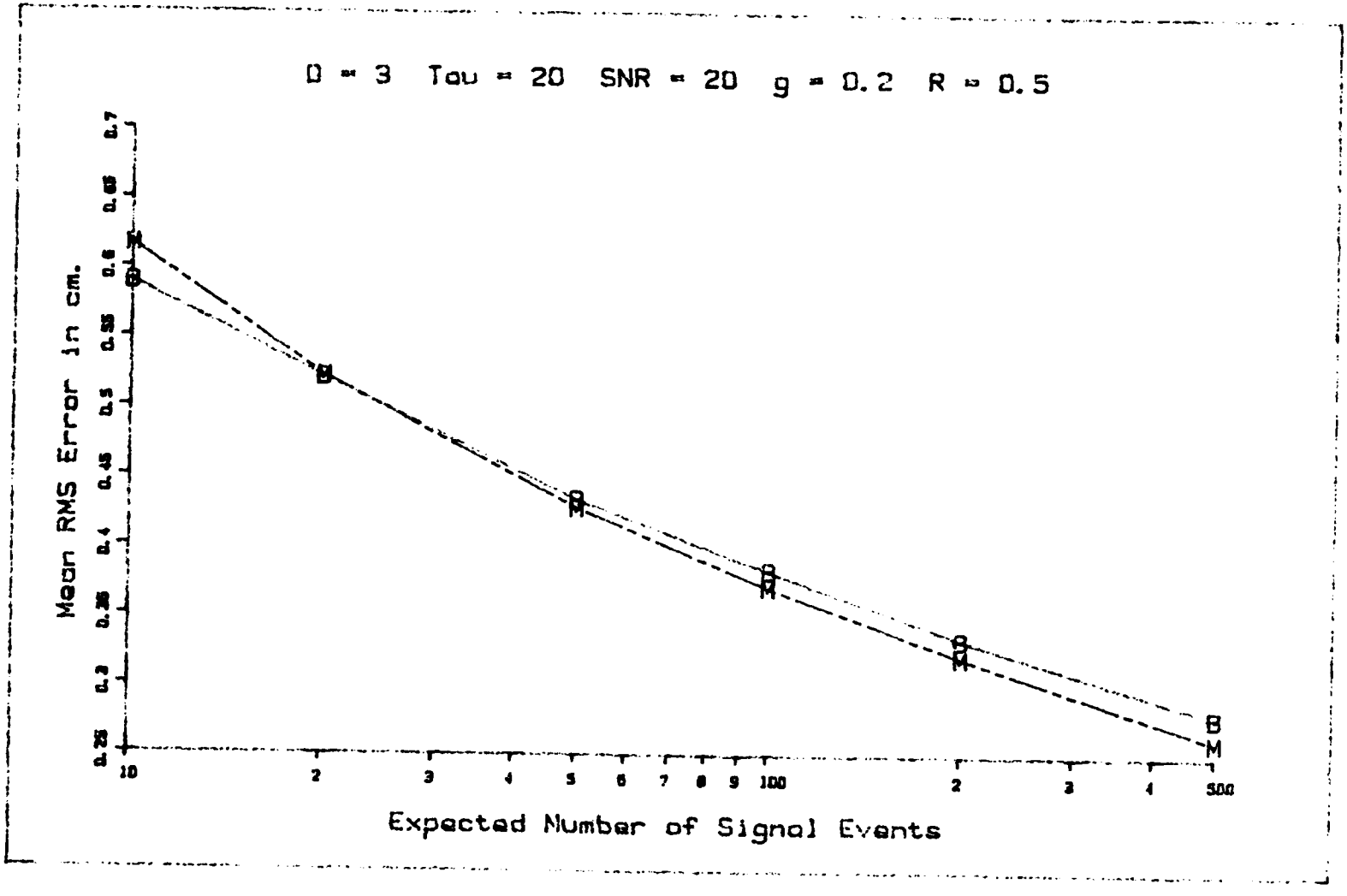


Figure 11'. Estimator Performance Versus Expected Number of Signal Events.

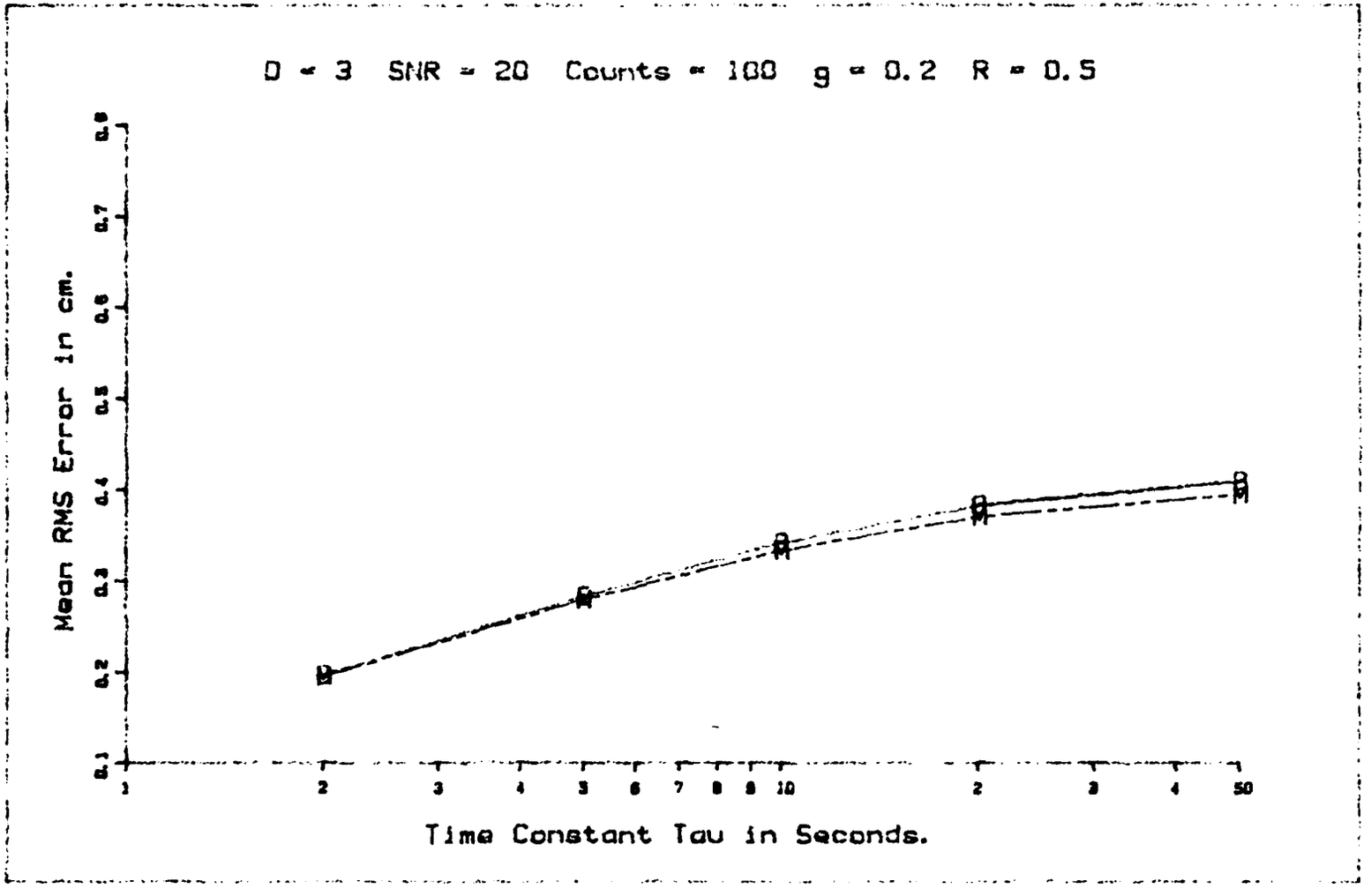


Figure 12'. Estimator Performance Versus Tau.

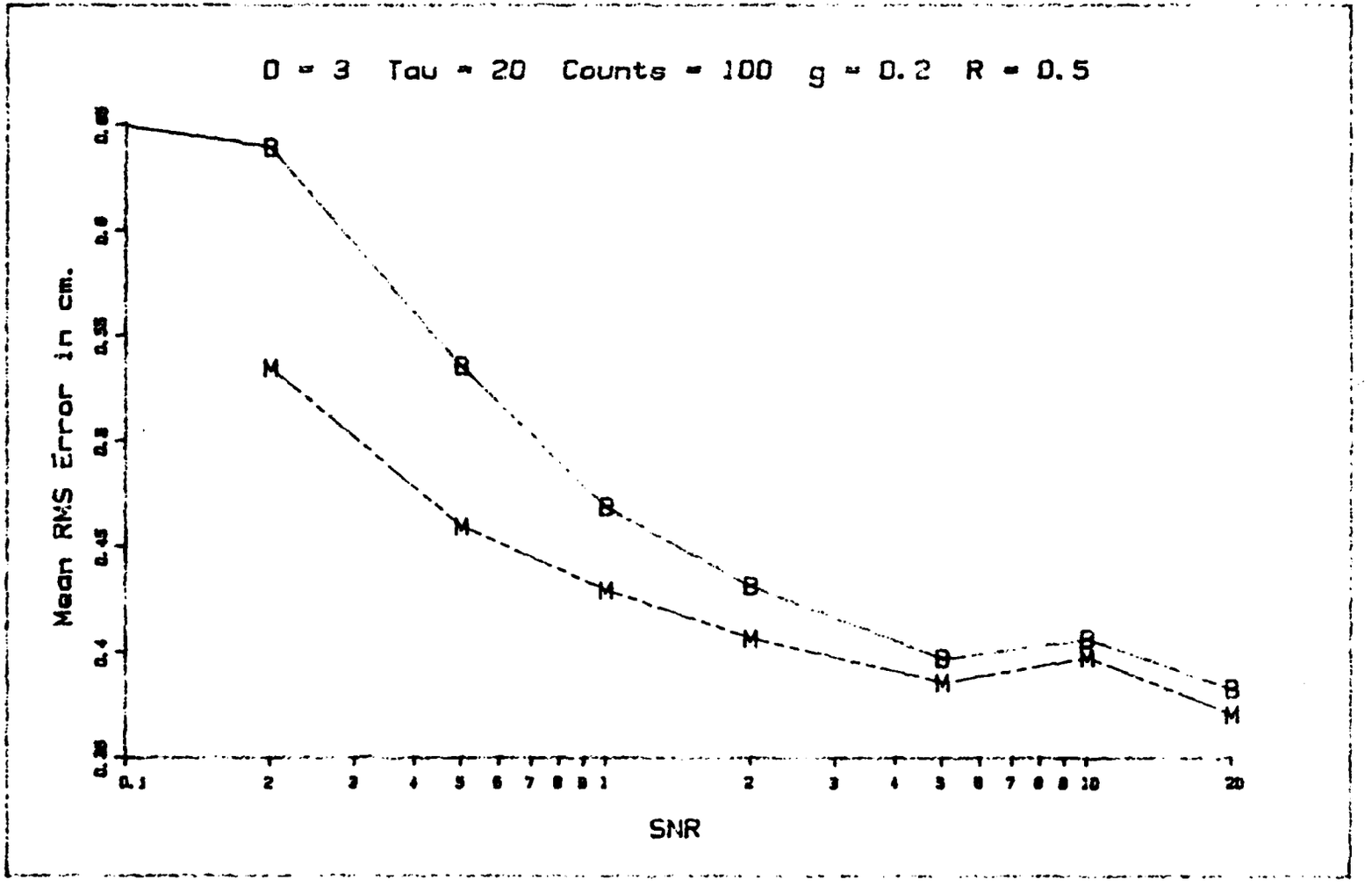


Figure 13'. Estimator Performance Versus Signal to Noise Count Ratio.

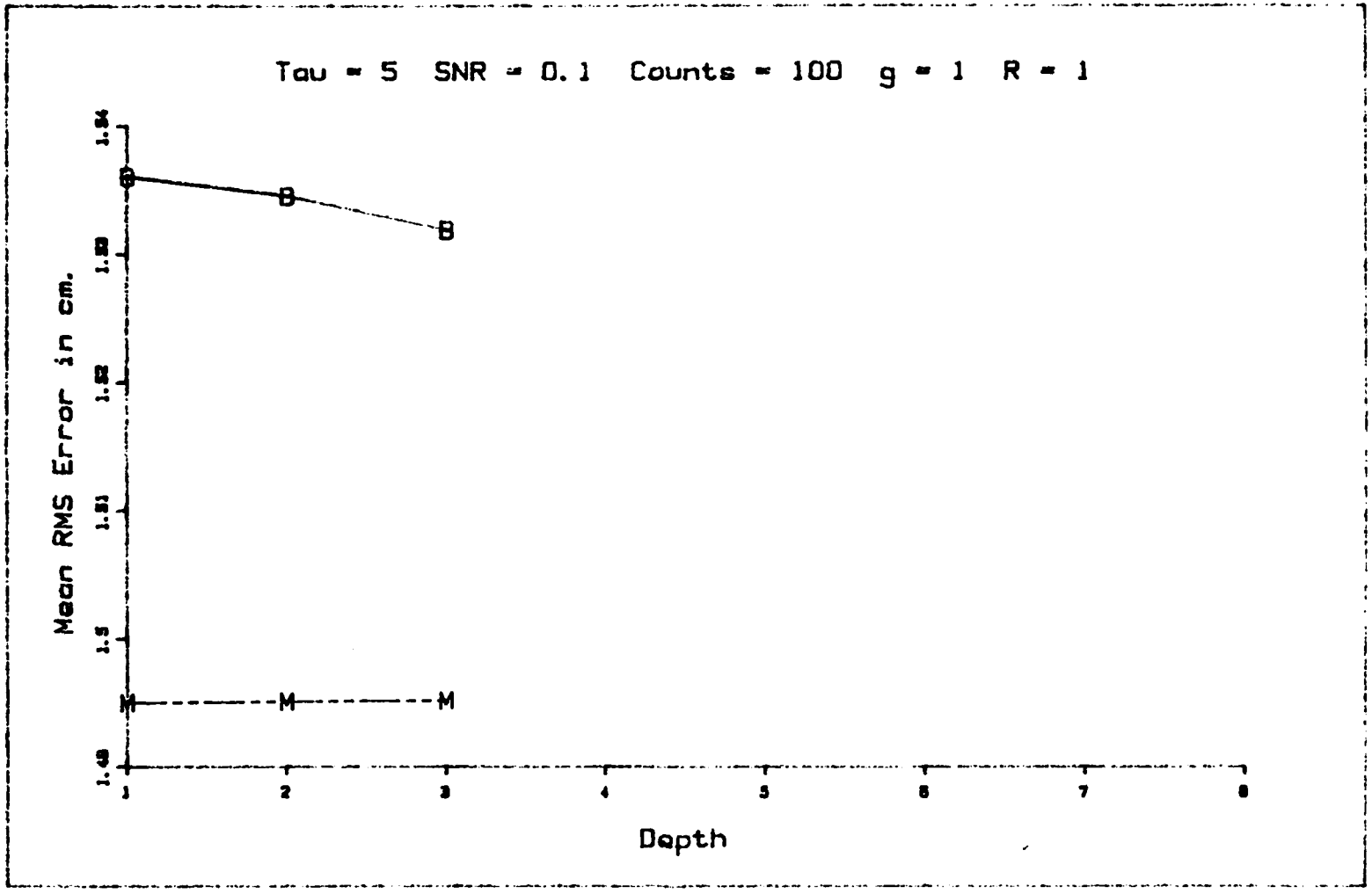


Figure 14'. Estimator Performance Versus Depth.

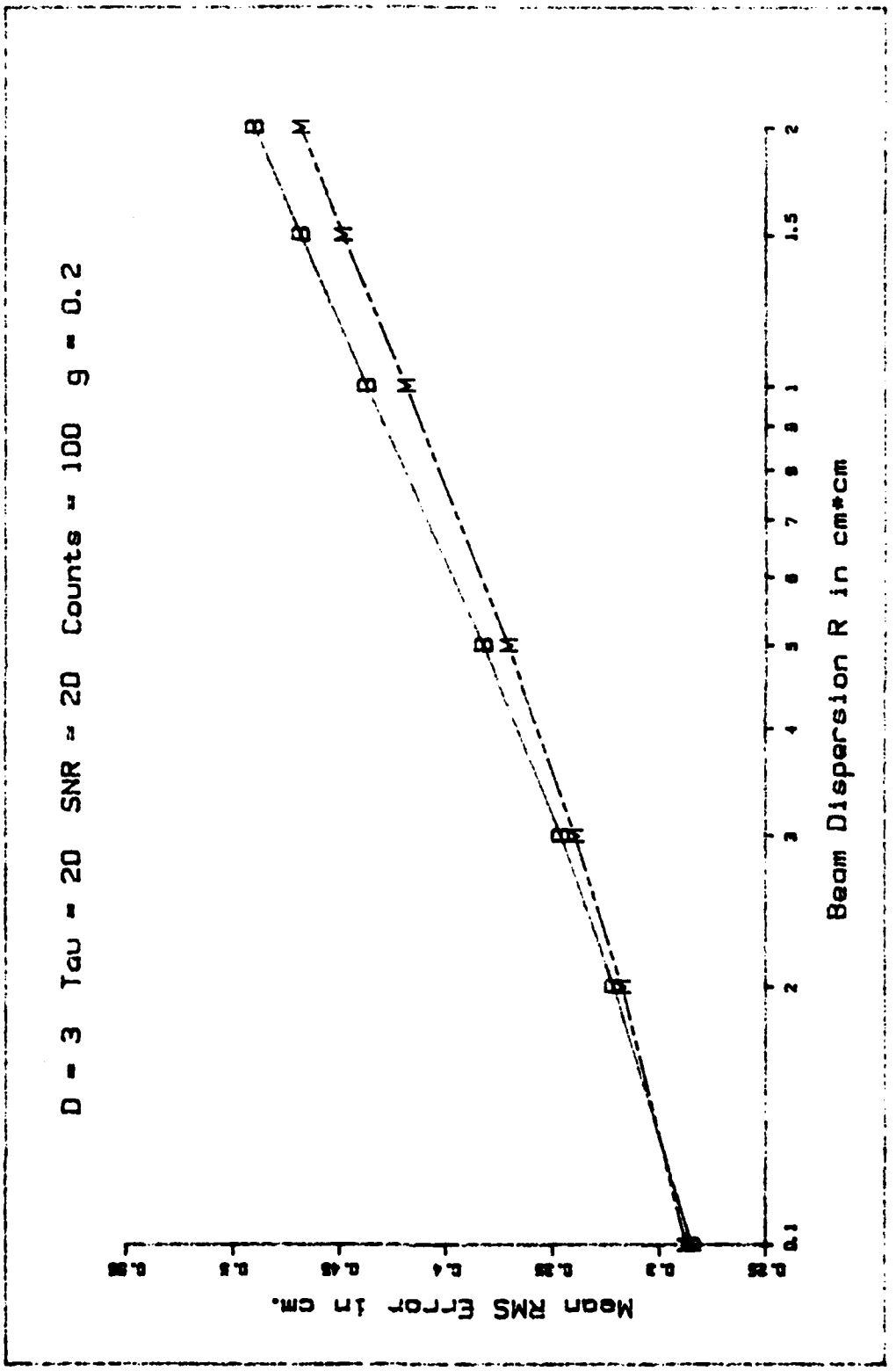


Figure 15'. Estimator Performance Versus Beam Dispersion.

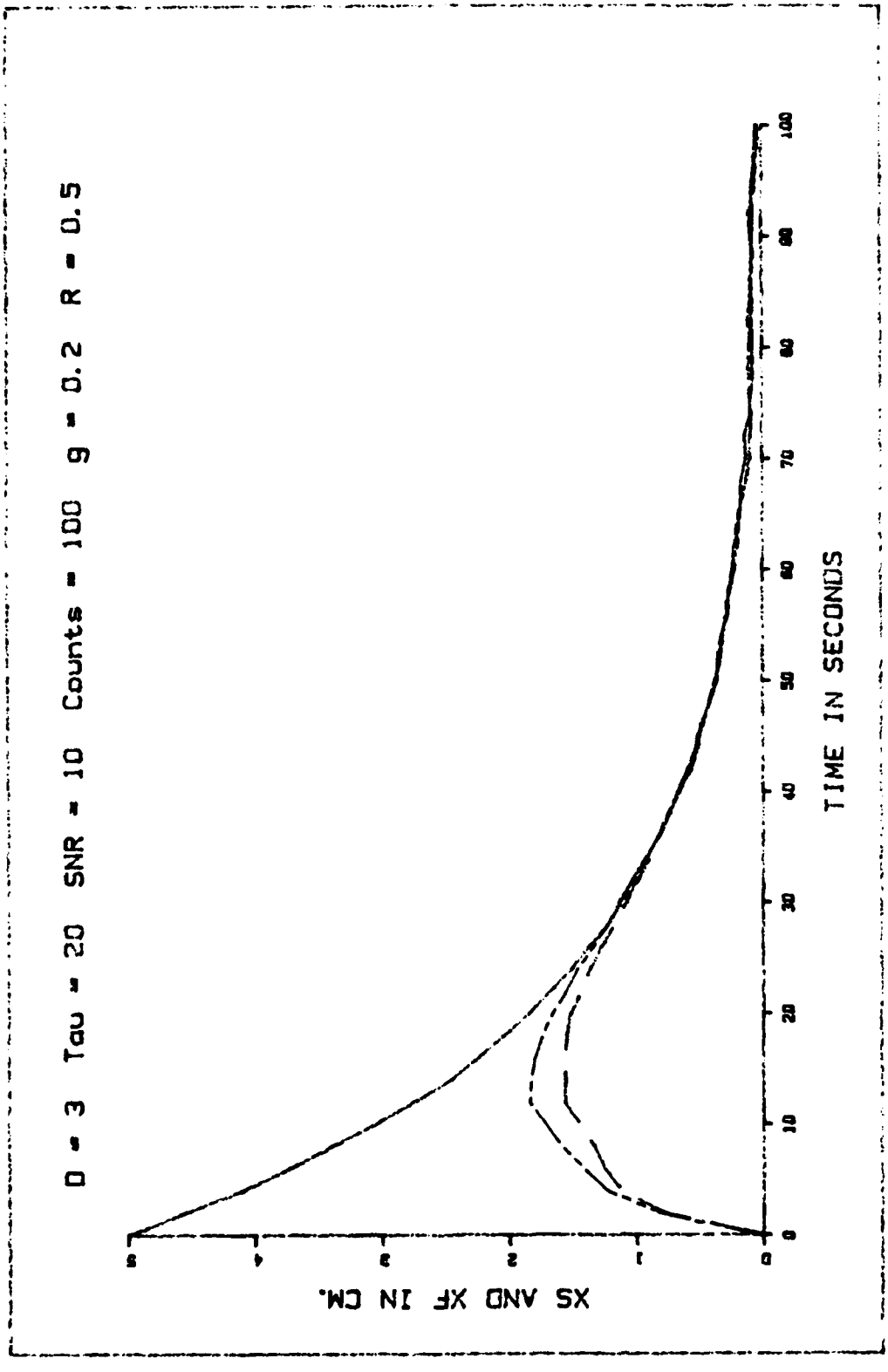


Figure 16'. Unsuccessful Acquisition.

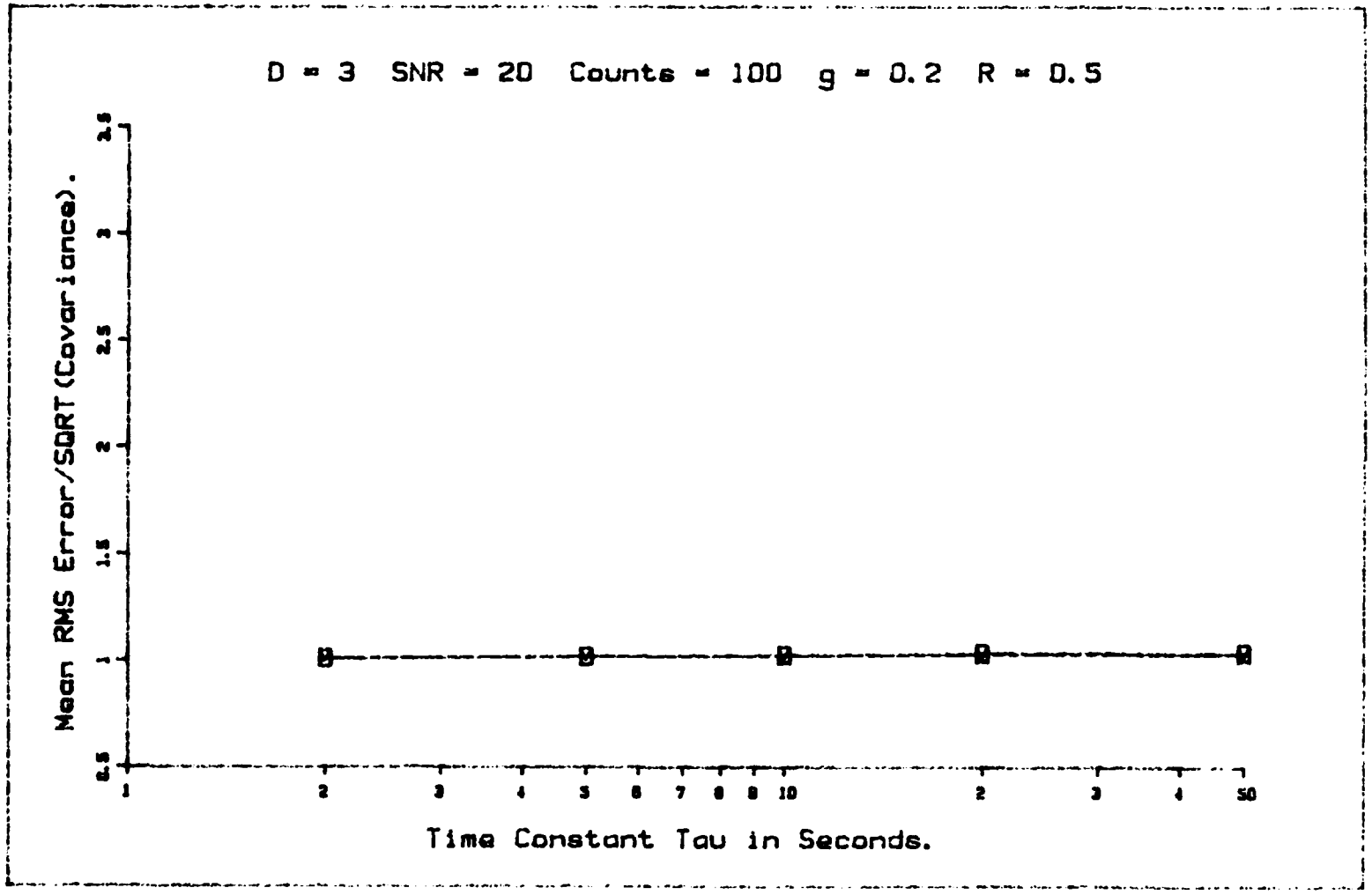


Figure 18'. Estimator Accuracy Versus Tau.

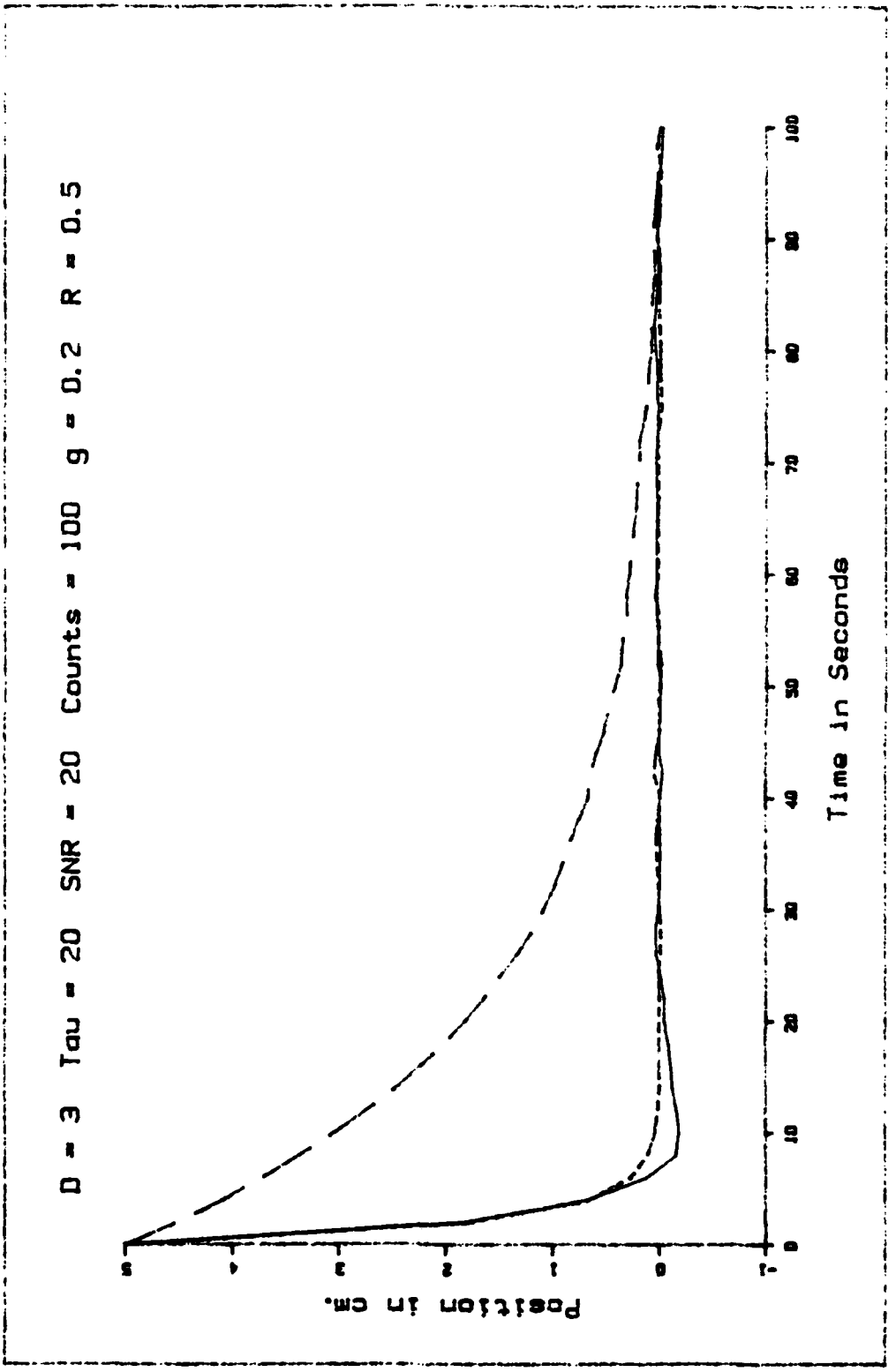


Figure 20'. Effect of Regulator on Time Response.

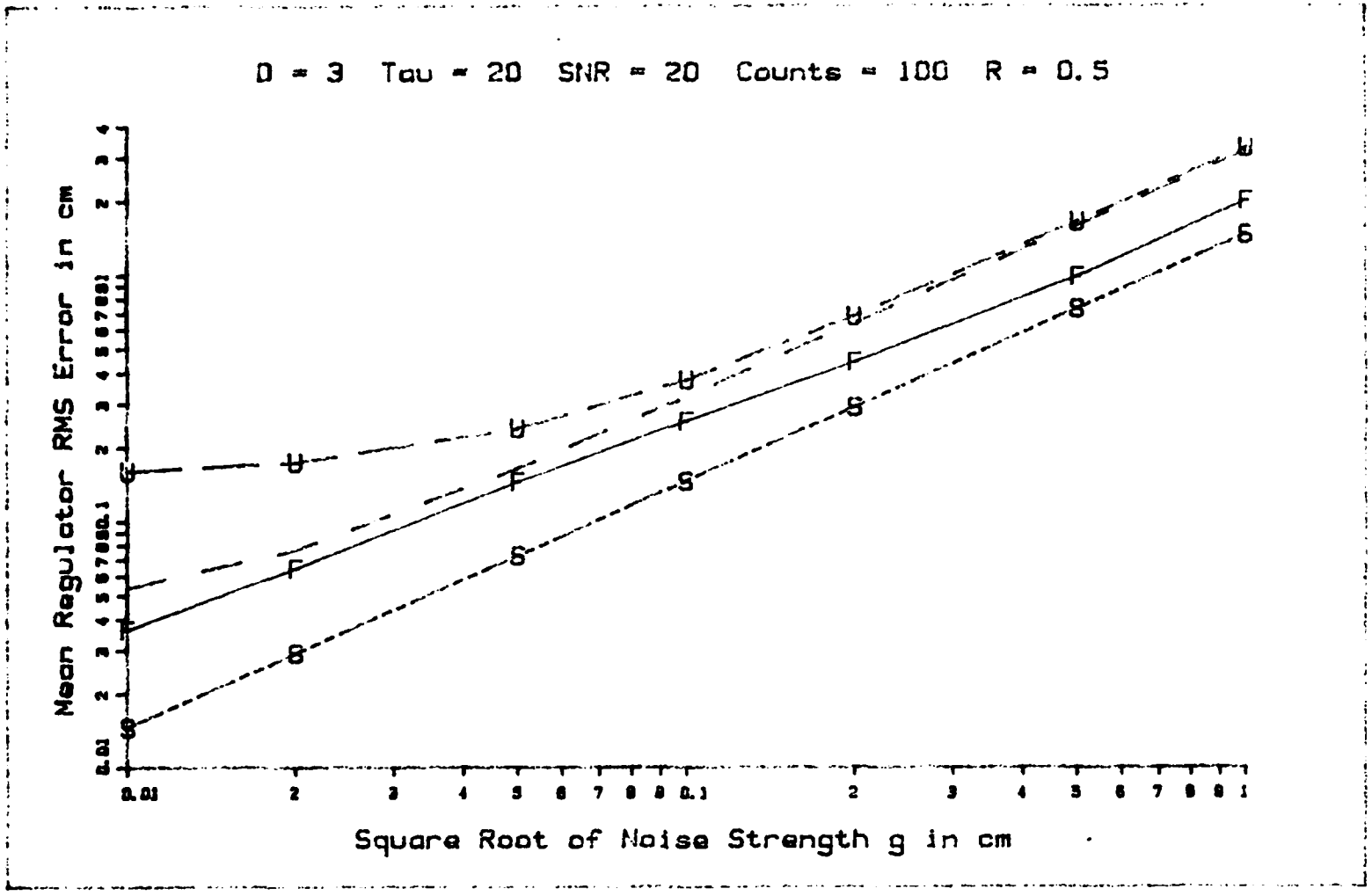


Figure 21'. Regulator Performance Versus Dynamic Model Noise.

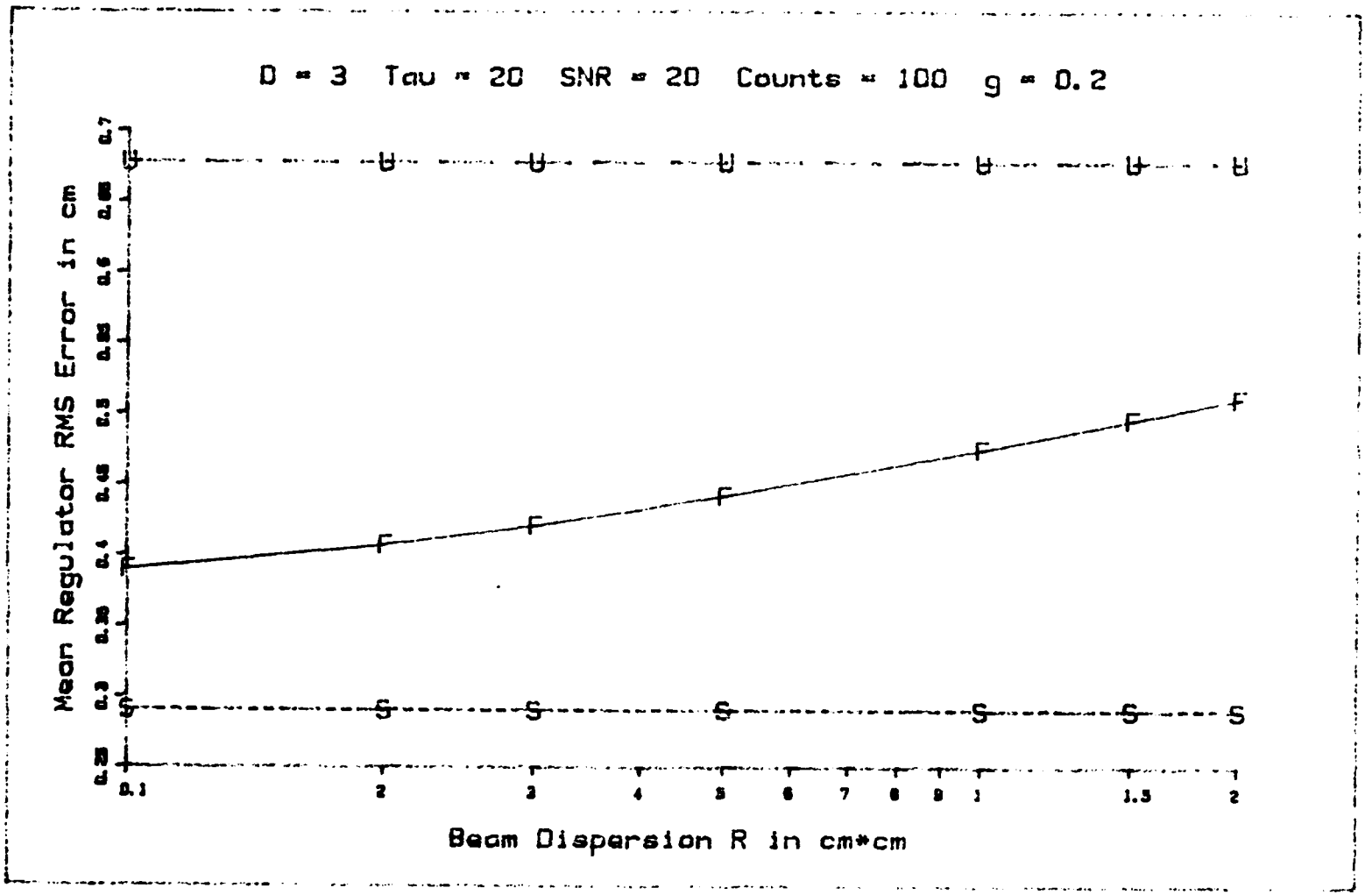


Figure 22'. Regulator Performance Versus Beam Dispersion.

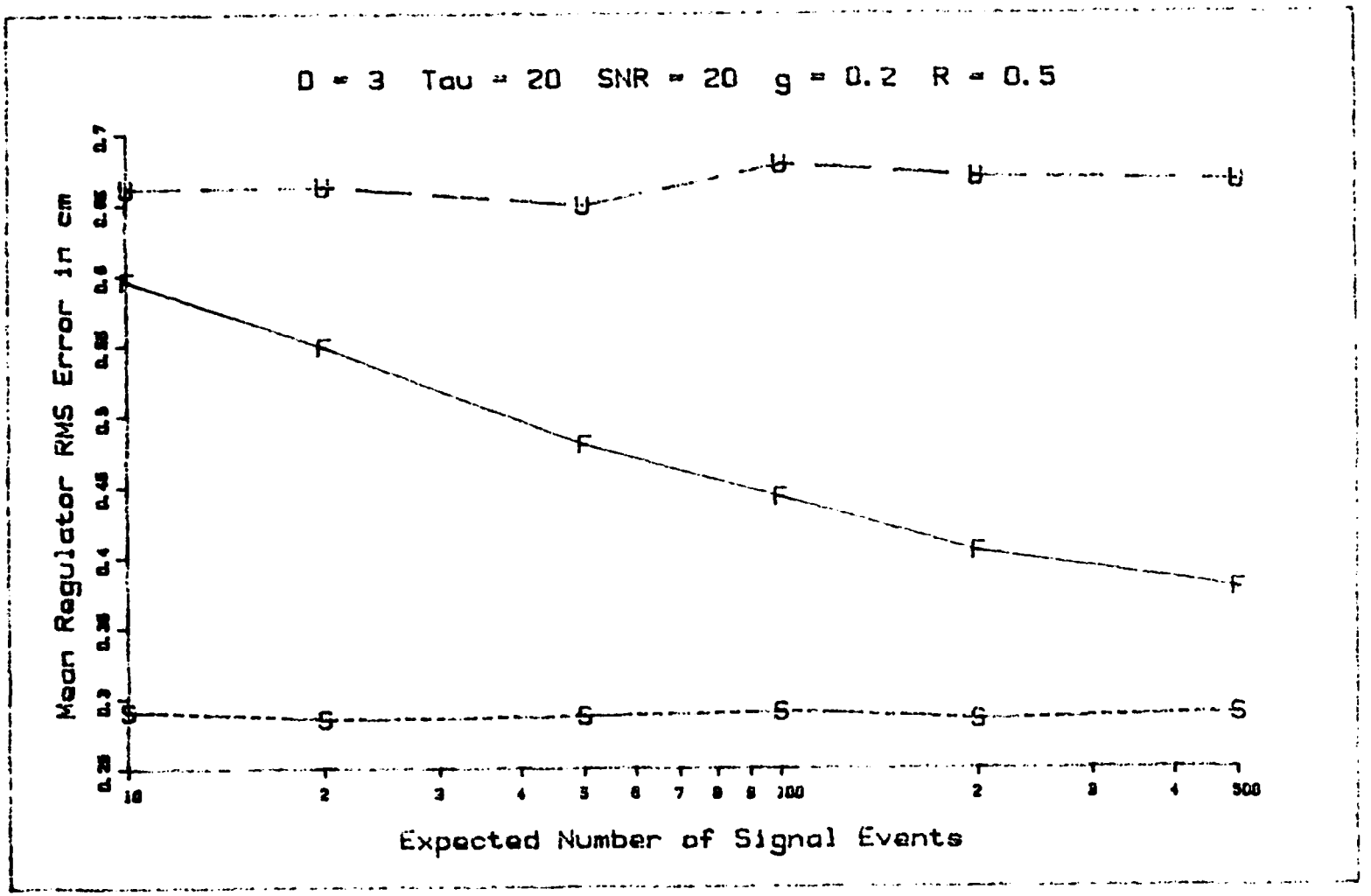


Figure 23'. Regulator Performance Versus Expected Number of Signal Events.

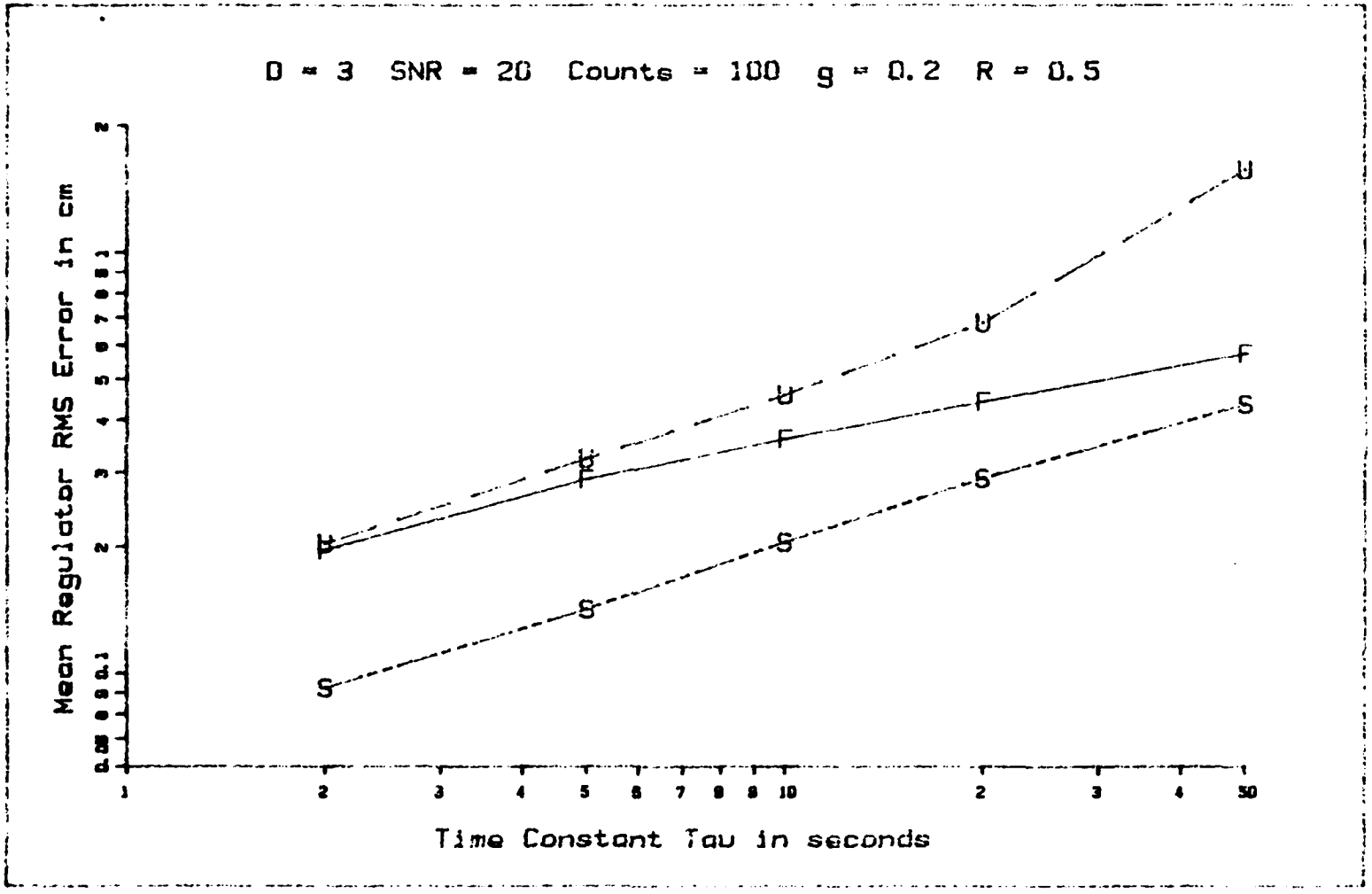


Figure 24'. Regulator Performance Versus Tau.

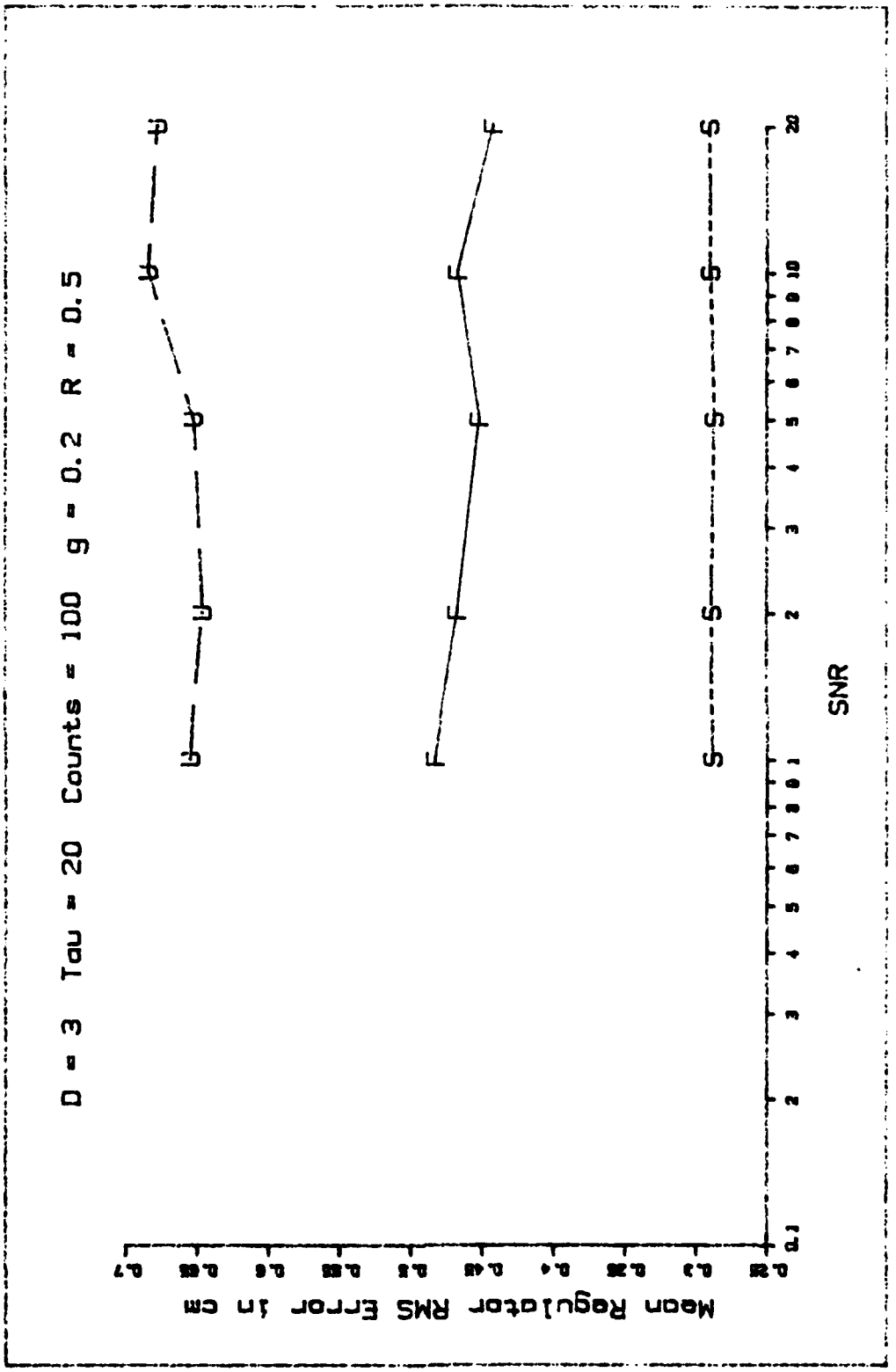


Figure 25'. Regulator Performance versus Signal to Noise Count Ratio.

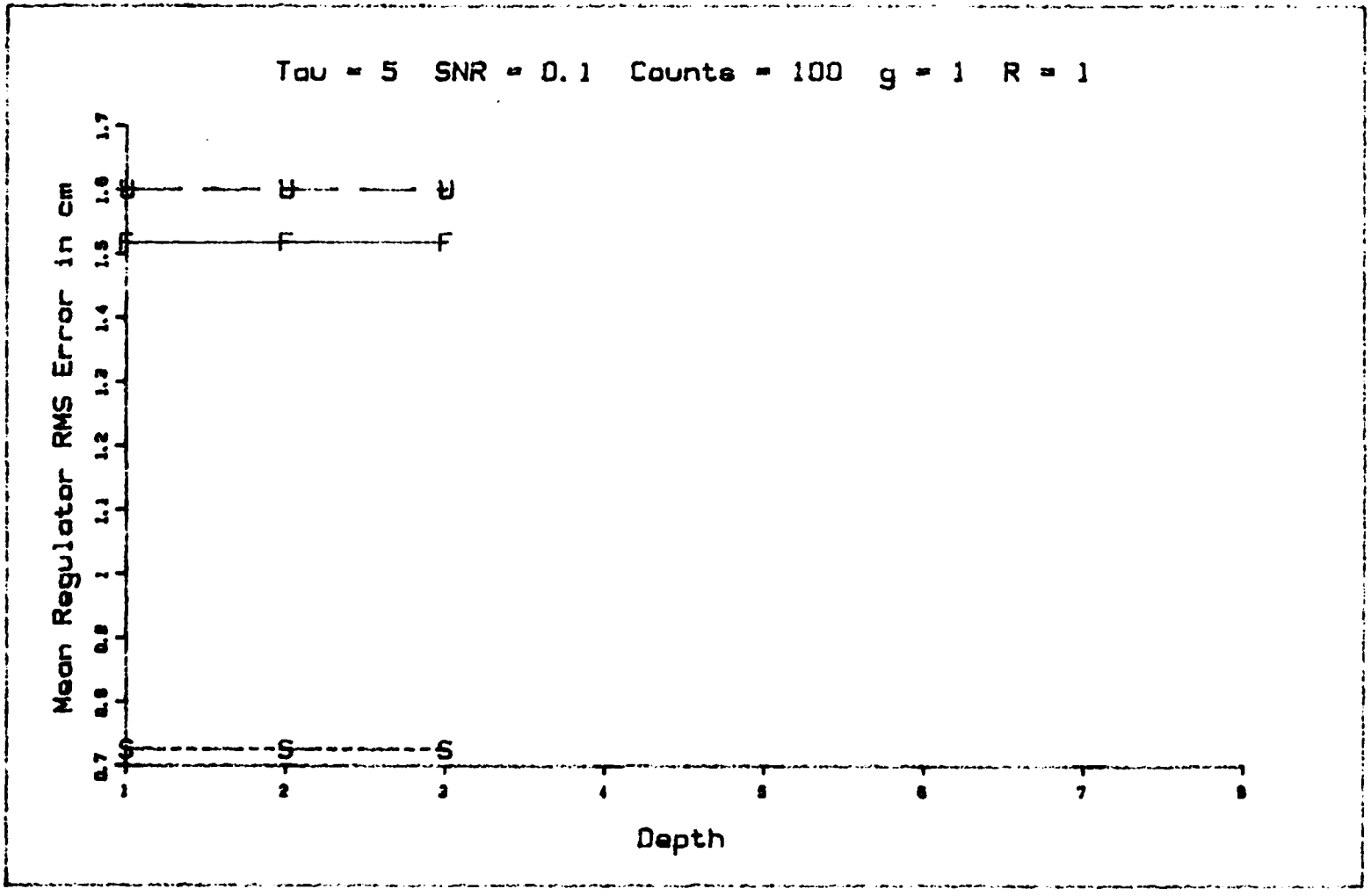


Figure 26'. Regulator Performance Versus Depth.

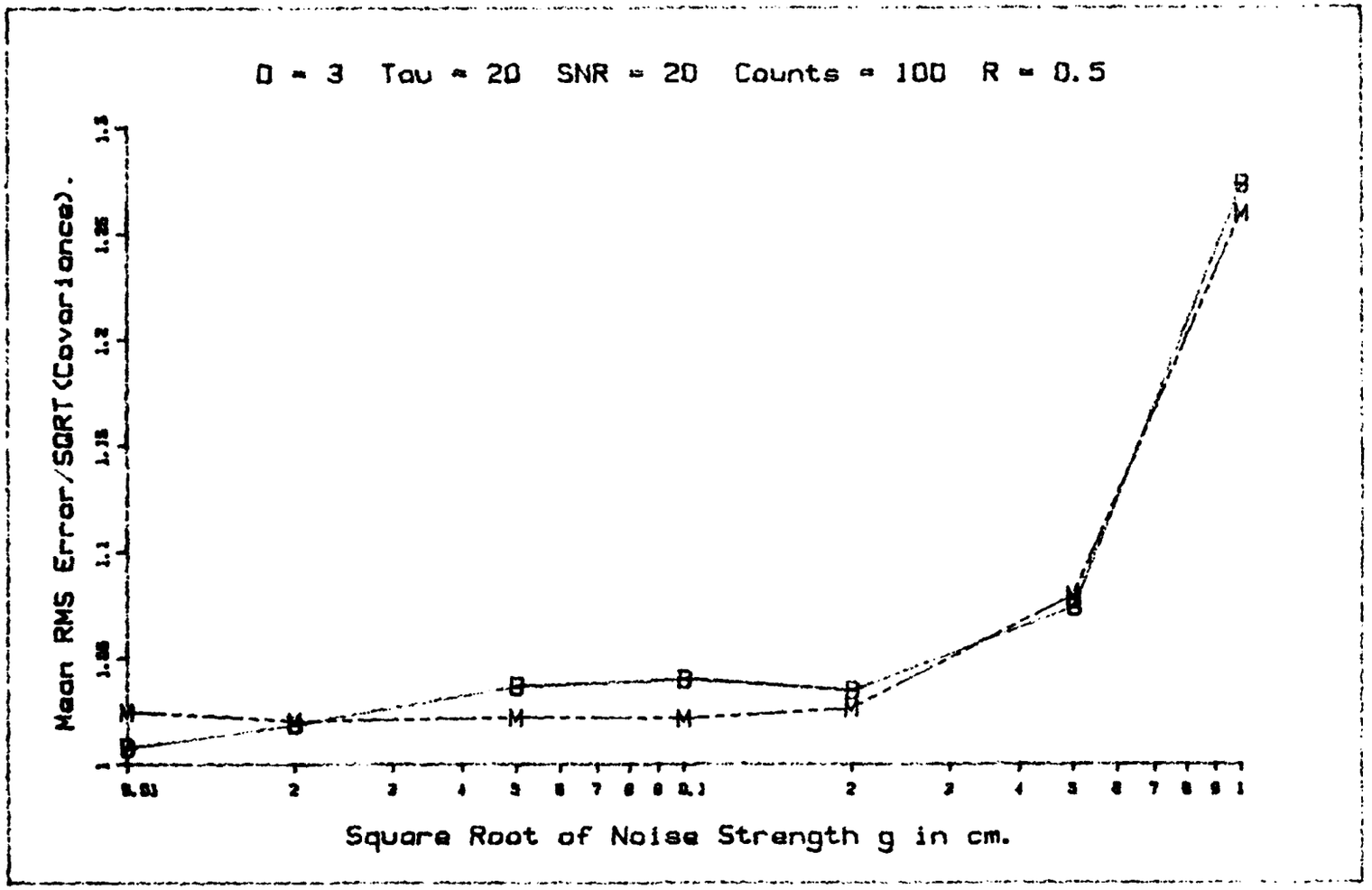


Figure 33'. Estimator Accuracy Versus Dynamic Model Noise.

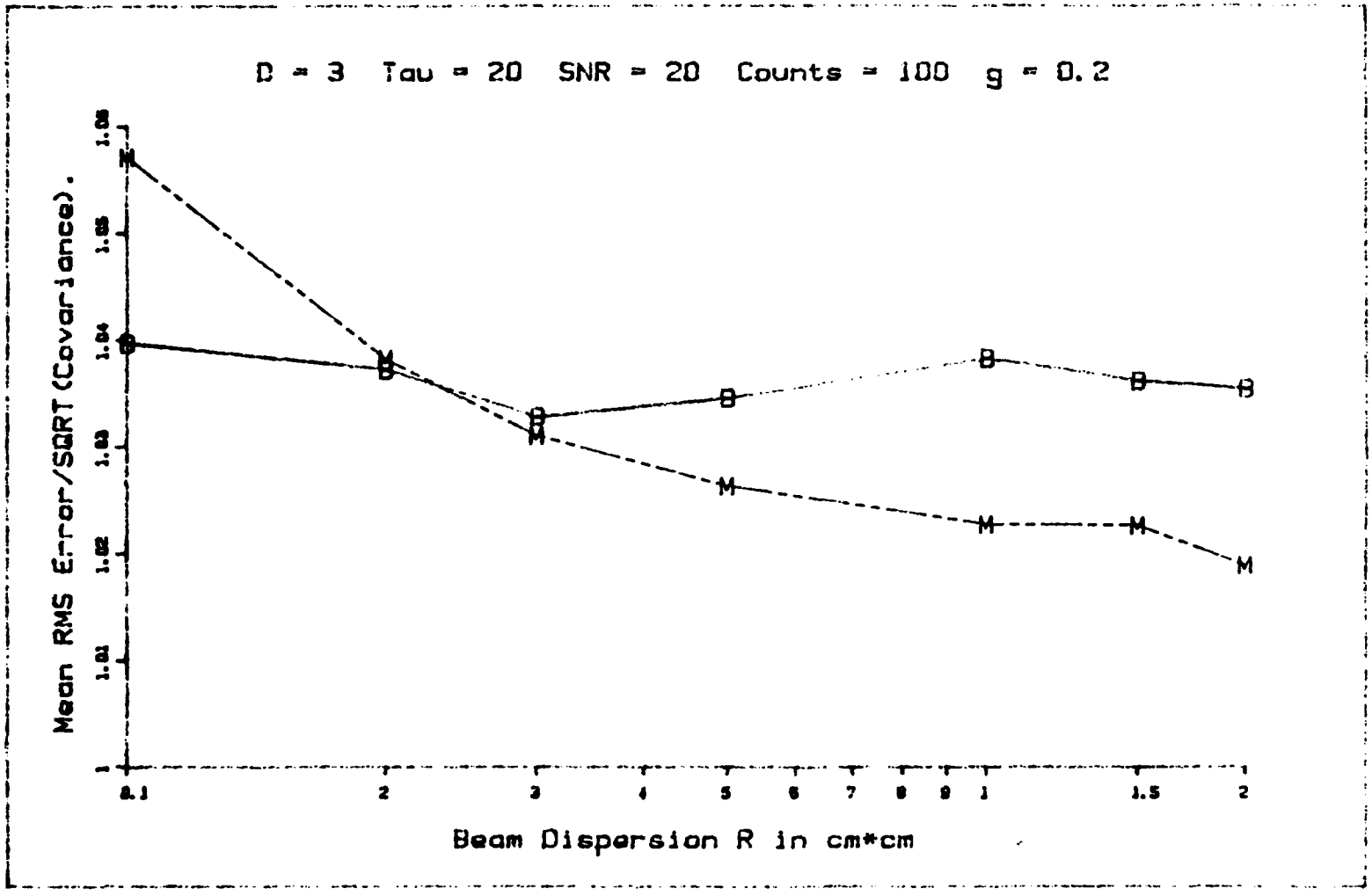


Figure 34'. Estimator Accuracy Versus Beam Dispersion.

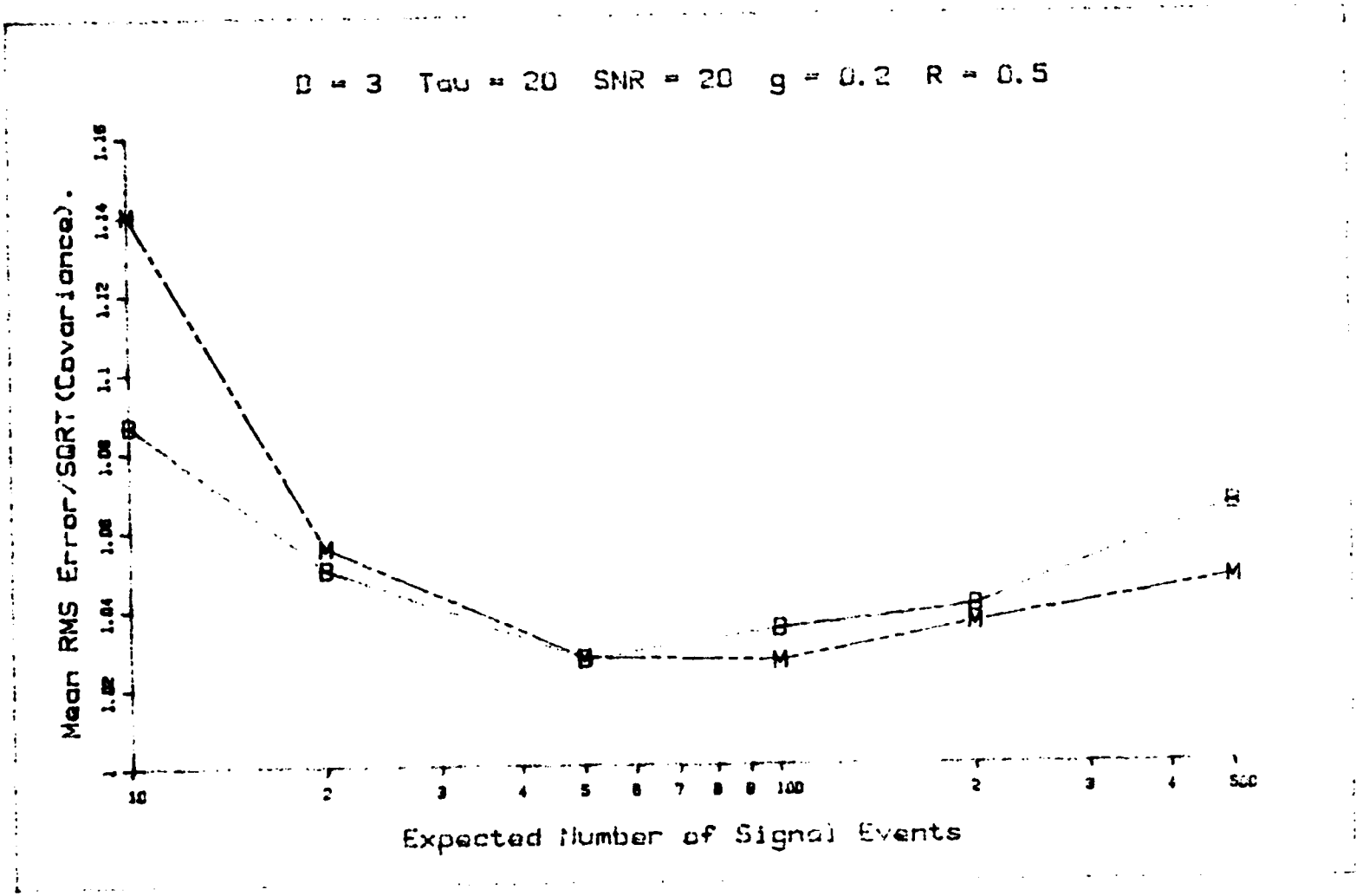


Figure 35'. Estimator Accuracy Versus Expected Number of Signal Events.

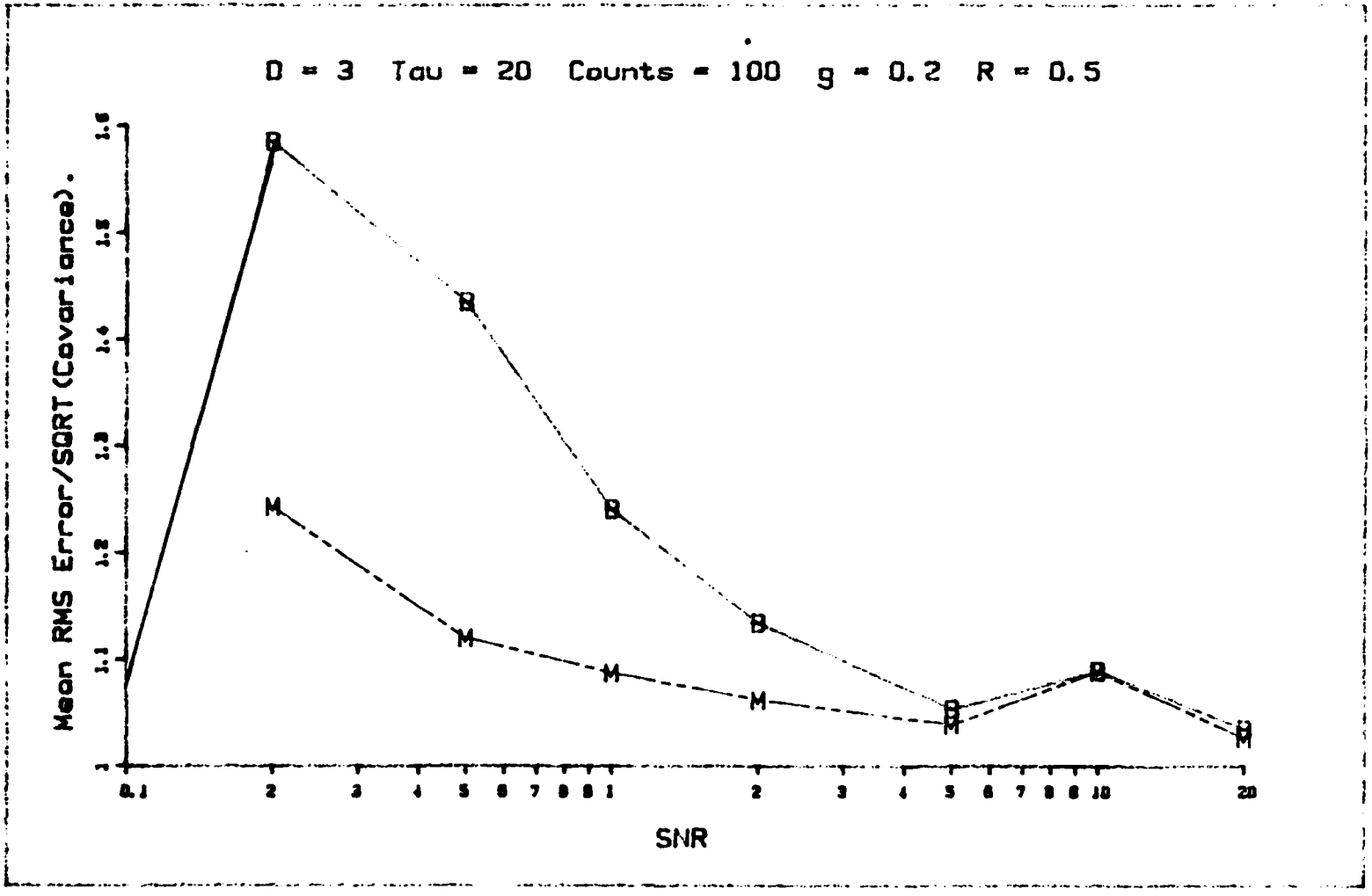


Figure 36'. Estimator Accuracy Versus Signal to Noise Count Ratio.

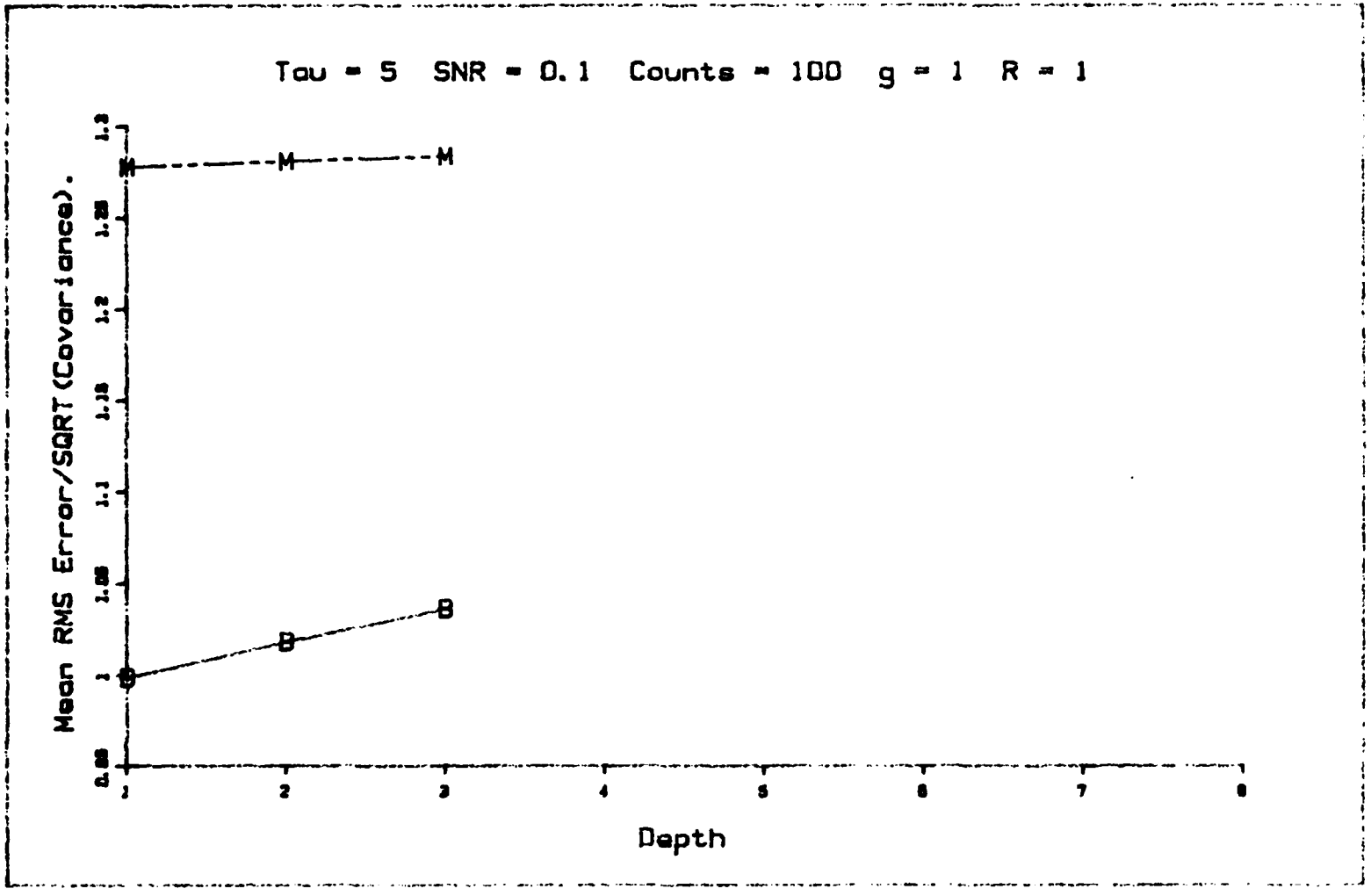


Figure 37'. Estimator Accuracy Versus Depth.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S): AFIT/GE/EE/83-73	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION / 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State and ZIP Code): Air Force Institute of Technology, Wright-Patterson AFB, Ohio 45433	7b. ADDRESS (City, State and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION / 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State and ZIP Code)	10. SOURCE OF FUNDING NOS. — PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT NO.
11. TITLE (Include Security Classification): See Box 19	

12. PERSONAL AUTHOR(S): William L. Zicker, B.S., 1 Lt, USAF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
MS Thesis	FROM _____ TO _____	1983 December	144

16. SUPPLEMENTARY NOTATION

Approved for public release: IAW AFR 190-17.

LYNN E. WOLAVER
Dean for Research and Professional Development
Air Force Institute of Technology (AIC)
Wright-Patterson AFB OH 45433

7Feb84

17. COSATI CODES		
FIELD	GROUP	SUB. GR.
09	04	
12	01	

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number): Point Process, Space-time Point Process, Multiple Model Adaptive Estimator, Adaptive Estimation, Optimal Filtering

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Title: POINTING OF TRACKING OF PARTICLE BEAM
Thesis Chairman: Peter S. Maybeck, PhD.

A problem is considered to determine the feasibility of using a Multiple Model Adaptive Estimator based on space-time point process observations, as part of a control loop. The estimator tracks the centroid of a one-dimensional Gaussian-shaped source of individual photo-electron events. The centroid is assumed to move dynamically as a first order Gauss-Markov process. Estimator performance for two implementations (using "best half" versus "merge" rationale for limiting the number of individual filters in the MMAE structure) is described in terms of steady-state root mean square (RMS) error evaluated as a function of six important system parameters. Estimates of the beam centroid are used by a regulator based on a (Cont.)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐	UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE NUMBER (Include Area Code)	22c. OFFICE SYMBOL
Peter S. Maybeck, PhD.	(513) 255-3576	AFIT/ENG

DD FORM 1473, 83 APR — EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

Linear system and Quadratic cost criteria (LQ) to determine the effectiveness of the method. For tracking purposes, a real-world target and a Kalman filter are provided based on a first order Gauss-Markov process model, and the controller is tested in this environment. Results indicate that the estimator provides a viable means of controlling the position of the centroid.

Subject Terms (Continued from Block 18): Optimal Control Theory, Regulator Design, Tracker, Certainty Equivalence, Snyder-Fishman Filter